

# Analítica de enseñanza y aprendizaje en cursos de programación

Teaching and learning analytics applied to programming courses

Juan C. Fiallos Quinteros<sup>1</sup>, Jovani A. Jiménez Builes<sup>1</sup>,  
Jhon W. Branch Bedoya<sup>1</sup>

<sup>1</sup> Universidad Nacional de Colombia, Colombia

jfiallos@unal.edu.co , jajimen1@unal.edu.co , jwbranch@unal.edu.co

**RESUMEN.** La enseñanza de la programación requiere del desarrollo de habilidades cognitivas de alto orden, lo que exige un gran esfuerzo por parte de estudiantes y profesores. Las altas tasas de fracaso académico indican que es necesario tomar medidas para revertir esta situación. La analítica de la enseñanza y el aprendizaje proporciona métodos, procesos y técnicas que permiten mejorar la calidad del proceso educativo. La investigación presenta una revisión sistemática de estudios en los que se aplican técnicas, métodos o procesos de análisis de la enseñanza y el aprendizaje en cursos de programación inicial en el contexto de la educación superior. El objetivo principal es identificar las principales perspectivas y tendencias en la analítica de enseñanza y aprendizaje aplicada a la programación y posibles temas de investigación.

**ABSTRACT.** Teaching programming requires the development of high-order cognitive skills, which demands a great effort from students and teachers. The high rates of academic failure indicate that it is necessary to take action to reverse this situation. Teaching and Learning Analytics provide methods, processes, and techniques that allow improve the quality of education process. The research presents a systematic review of studies in which techniques, methods or processes of teaching and learning analysis are applied in initial programming courses in the context of higher education. The main purpose is to identify the main perspectives and trends in teaching and learning analytics applied to programming and potential research topics.

**PALABRAS CLAVE:** Analítica, Enseñanza, Aprendizaje, Programación, Educación.

**KEYWORDS:** Analytics, Teaching, Learning, Programming, Education.

## 1. Introducción

Uno de los problemas que afronta al momento la educación superior son las altas tasas de deserción y repitencia de los alumnos, principalmente en cursos iniciales. En Latinoamérica menos de la mitad de los estudiantes que empiezan una carrera pueden terminarla. En Colombia existe un alto índice de deserción estudiantil en la educación superior de más del 45% (Mineducación, 2012); además, alrededor del 36% de los estudiantes que desertan lo hacen al final del primer año (Ferreira, Avitabile, Botero, Haimovich & Urzúa, 2012), esto ha motivado en la última década la discusión en torno al tema y la necesidad de generar mecanismos de soporte al proceso de enseñanza aprendizaje. Para el caso de cursos iniciales de programación se presenta una problemática similar ya que la literatura reporta tasas mundiales de aprobación de apenas el 67.7% (Watson, 2014). Esto, además de los costos educativos y sociales, genera problemas productivos debido a la creciente demanda de profesionales en tecnologías de información que no se ve satisfecha (Casey & Azcona, 2017).

La incorporación de Tecnologías de Información y Comunicación (TIC) en el ámbito educativo (Morales Salas et al., 2019; Infante Moro et al., 2020a, 2020b, 2021a, 2021b), a través de diversas aplicaciones como son los Sistemas de Gestión del Aprendizaje (LMS, por sus siglas en inglés), tutores inteligentes, ambientes virtuales de aprendizaje, herramientas de calificación automática, entre otros, ponen a disposición una enorme cantidad de datos acerca del proceso de enseñanza-aprendizaje.

La analítica aplicada a la educación puede sacar provecho de este gran cúmulo de información, a través de tareas como la medición, recolección, análisis y reporte de datos; de esta manera extraer conocimiento sobre los estudiantes, profesores y su contexto, con el objetivo de entender y optimizar el proceso de enseñanza aprendizaje. Lo anterior puede realizarse mediante el uso de herramientas y técnicas, como la estadística, análisis visual, el análisis de redes sociales y analítica predictiva (Chatti, Dyckhoff, Schroeder & Thüs, 2012).

La Analítica del Aprendizaje (LA) abarca la investigación educativa desde la perspectiva del estudiante; mientras que, la Analítica de la Enseñanza (TA) aborda la investigación desde la perspectiva del profesor. La Analítica de la Enseñanza y del Aprendizaje (TLA) propone un enfoque integral, holístico, en el que los conocimientos generados por los métodos y herramientas de LA puedan traducirse de manera significativa hacia impulsar la investigación docente para mejorar su práctica, capturada a través de los métodos y herramientas de TA (Sergis & Sampson, 2017).

El objetivo de este trabajo busca dar respuesta a las siguientes preguntas de investigación:

PI1 ¿Cuáles son las principales perspectivas y tendencias en Analítica de la Enseñanza y la Analítica del Aprendizaje de trabajos científicos que tratan el dominio de programación en educación superior?

PI2 ¿Cuáles son los potenciales temas de investigación en la integración de Analítica de la Enseñanza y el Aprendizaje para cursos de programación en educación superior?

Las subsiguientes secciones se encuentran organizadas de la siguiente manera: en la Sección II se presenta una definición de términos y conceptos sobre EDM, LA y TA; en la Sección III se describe el protocolo de investigación; seguidamente, en la Sección IV se analizan y se discuten los resultados obtenidos; y, finalmente, en la Sección V se responden las preguntas de investigación planteadas.

## 2. Revisión de la literatura

La disponibilidad de grandes volúmenes de información permitió desarrollar métodos, procedimientos y metodologías que permiten una comprensión de esa información y a partir de aquello tomar mejores decisiones. De esa manera las TIC y las ciencias relativas a la toma de decisiones, como la psicología y ciencia del comportamiento, la matemática, la estadística y la economía-econometría aportaron a la conformación de disciplinas que habilitan el “uso extensivo de datos, estadística y análisis cuantitativo, modelos predictivos y explicativos, y la gestión basada en hechos para orientar acciones y decisiones” (Mortenson, Doherty &



Robinson, 2014).

El rendimiento académico de los estudiantes universitarios constituye un factor imprescindible en el abordaje del tema de la calidad de la educación superior, debido a que es un indicador que permite una aproximación a la realidad educativa (Garbanzo, 2007). El proceso de aprendizaje se ve afectado por distintos factores como son los institucionales, pedagógicos, sicosociales, sociodemográficos, entre otros. Las notas obtenidas, como un indicador que certifica el logro alcanzado son un indicador preciso y accesible para valorar el rendimiento académico (Mora, 2015).

Las teorías de aprendizaje nos posicionan sobre cómo concebimos el proceso de enseñanza aprendizaje, cómo establecemos estrategias pedagógicas, desarrollamos recursos y evaluamos este proceso (Greller & Drachsler, 2012). Las teorías de aprendizaje han aportado al desarrollo de enfoques pedagógicos que determinan la concepción de éxito en el proceso de enseñanza aprendizaje (Knight, Buckingham & Littleton, 2014).

En la investigación educativa existen dos grandes tendencias: la Analítica del Aprendizaje (Learning Analytics, LA) y la Analítica de la Enseñanza (Teaching Analytics, TA). La Analítica del Aprendizaje se define como "la medición, recolección, análisis y reporte de datos acerca de estudiantes y su contexto, con el objetivo de entender y optimizar el aprendizaje y su entorno" (Long & Siemens, 2011); mientras que, la Analítica de la Enseñanza (Teaching Analytics, TA) se refiere al análisis y mejora de la práctica de la enseñanza, esta última se enmarca en la "investigación docente" que se define como un "proceso continuo de investigación, reflexión y mejora de la práctica de enseñanza, basada en la recolección, análisis e interpretación de diversos datos educativos" (Sergis & Sampson, 2017).

Por separado, la Analítica de la Enseñanza y la Analítica del Aprendizaje se enfoca cada una en el análisis desde la perspectiva del profesor y del estudiante de manera independiente; esto representa una dificultad debido a que conciben y resuelven un problema de manera parcial. Por lo anterior se ha propuesto nuevo enfoque de análisis integral conocido como Analítica de la Enseñanza y el Aprendizaje con el objetivo de integrar en un solo enfoque que atienda la problemática y proporcione un marco de trabajo en el que los conocimientos generados por los métodos y herramientas de la Analítica del Aprendizaje puedan traducirse de manera significativa para impulsar la investigación de los docentes para mejorar su práctica, capturada a través de los métodos y herramientas de la Analítica de la Enseñanza (Sergis & Sampson, 2017).

### 3. Metodología

El objetivo de la revisión consistió en identificar los estudios que aplicaran técnicas de analítica, ya sea con el enfoque de enseñanza o aprendizaje en cursos formales de programación o ciencias de la computación en el ámbito de la educación superior, cuya publicación fuera partir del año 2013. Para cumplir con este propósito se utilizó como fuente de investigación principal la base de datos Scopus, ya que es la mayor base de datos de resúmenes y contiene la mayor cantidad de revistas reconocidas, representa aproximadamente un 80% de las publicaciones internacionales revisadas por especialistas. La cadena de búsqueda aplicada fue:

("data mining" OR "machine learning" OR analytics) AND (cs1 OR "computer science" OR programming) AND (teaching OR learning) AND (universit\* OR college OR "higher education").

En la búsqueda se incluyen los títulos, resúmenes y palabras clave de los documentos recuperados hasta octubre de 2019. Los criterios de inclusión fueron que los artículos provengan de una revista científica, conferencias internacionales, capítulos de libros, que refieran el uso de alguna técnica, proceso o metodología analítica aplicada el proceso de enseñanza aprendizaje en cursos formales de programación en educación superior.

No se incluyeron artículos que presentan estudios relacionados a niveles de educación primaria ni

secundaria, tampoco se incluyen aquellos que presentan resúmenes de conferencias ni los que abordan el estudio de cursos tipo MOOC.

Como resultado de la aplicación de la cadena de búsqueda se obtuvo 619 documentos que luego de una revisión de los resúmenes a los que se aplicaron los criterios de inclusión y exclusión y como resultado quedaron 121 documentos. Estos fueron analizados en detalle y se depuraron hasta quedar en un número de 55 que se corresponden con los criterios de la revisión.

## 4. Resultados

En esta sección se presentan los resultados de la revisión, se responden las preguntas de investigación y se presentan los principales hallazgos.

RQ1 ¿Cuáles son las principales perspectivas y tendencias en analítica de la enseñanza y el aprendizaje de trabajos científicos que tratan el dominio de programación en educación superior?

Para responder esta pregunta se decidió adaptar la taxonomía propuesta por Peña-Ayala (2018), la cual establece categorías para la Analítica del Aprendizaje y se ha incluido una categoría adicional para la Analítica de la Enseñanza. Se ha tomado el área que propone dicha taxonomía que aborda las aplicaciones que tiene la Analítica del Aprendizaje y describe las funcionalidades, el análisis del estudiante y los recursos que se generan para la investigación; y, finalmente los factores implícitos que incluyen marcos de trabajo, guías, análisis éticos y legales, así como paradigmas de aprendizaje que se aplican a la investigación de la TLA. Si bien todos los estudios no solamente abordan una temática definida, en el presente análisis se agrupa su principal objetivo de investigación. A continuación, se describe cada categoría en la que encontraron investigaciones y se reseñan los trabajos encontrados.

### 4.1. Funcionalidades

#### 4.1.1. Predicción de estudiantes en riesgos

Una de las principales funcionalidades que tiene la Analítica del Aprendizaje es la identificación de estudiantes en riesgo de fracaso académico, para lo cual seleccionan variables relacionadas con el estudiante, mismas que se procesan con técnicas y algoritmos de analítica y minería de datos que permiten definir modelos para extraer patrones que sirven para identificar estudiantes en riesgo de fallar en el curso. La característica fundamental de esta funcionalidad es que permita la identificación temprana de los estudiantes en riesgo para que se pueda dar una intervención y así mitigar ese riesgo. Estos estudios trabajan una variable objetivo de tipo dicotómica; es decir, si aprobará o no el curso.

Para identificar los estudiantes en riesgo el estudio de Azcona & Smeaton (2017) elaboró un modelo predictivo en el que se utilizaron los registros de interacciones de los estudiantes de años anteriores con los recursos en línea y los reportes de progreso en los laboratorios de programación. Las predicciones se realizaron semanalmente para el curso donde el profesor recibía un reporte de los estudiantes con mayor posibilidad de reprobación el siguiente examen en laboratorio, a los que se ofrecía asistencia durante las sesiones de laboratorio. Los resultados indican que hubo una notable mejoría en los estudiantes que recibían una asistencia. En Ahadi (2016) se analiza la captura del código para determinar los errores en las tareas de programación. En Azcona, Hsiao y Smeaton (2019) se realiza la predicción automáticamente al momento de enviar una tarea, lo que permite entregar una retroalimentación con base en características sociodemográficas de los estudiantes, el desempeño en las tareas de programación y los registros de actividad. Las predicciones se realizan semanalmente. Costa et al. (2017) evalúa cuatro técnicas de minería de datos que utilizan información sociodemográfica, las calificaciones de evaluaciones semanales y la actividad realizada en la plataforma, analiza la efectividad del preproceso de datos y el ajuste de parámetros de cada técnica, lo que permite identificar a los estudiantes en riesgo a mitad del curso. Por su parte, McGowan et al. (2020) utiliza el análisis multimodal como atributos para la identificación de los estudiantes en riesgo, la frecuencia de repetición de visualizaciones



de videos de clase, asistencia, posición que ocupan los estudiantes en el aula y la frecuencia cardíaca a través del uso de dispositivos electrónicos de vestimenta (wearables).

#### 4.1.2. Predicción de desempeño

Otra funcionalidad es la predicción del desempeño de los estudiantes a través de factores relacionados con el desempeño propio del estudiante. En este sentido, para Sagar, Gupta y Kaushal (2016) el propósito fue predecir el desempeño en programación y analizar los impactos de varios atributos; establecen factores relacionados con los estudiantes como la habilidad de cada estudiante en programación y la cantidad de envíos. En Badr et al. (2016) se plantea la predicción del rendimiento académico del estudiante en el curso, a través de las calificaciones de otras asignaturas (inglés y matemáticas), para lo cual realiza la clasificación con el algoritmo association rules y permite predecir correctamente con un 52.94% de precisión. Otra característica usada en la tarea de predicción es el comportamiento del trabajo en equipo en el desarrollo de un programa. En Petkovic et al. (2015) se propone el uso de datos como medidas de actividad de trabajo en equipo (correos entre el equipo) y observaciones de los instructores, para esto elabora un modelo a través del algoritmo random forest que permite identificar factores para predecir las notas finales del desempeño del equipo de trabajo y la calidad del producto.

Watson, Li y Godwin (2013) presentan un enfoque para predecir el desempeño del estudiante con base en los datos de acceso al sistema y el comportamiento en programación. Para caracterizar el comportamiento en programación de los estudiantes se desarrolló un algoritmo (Watwin) que establece una puntuación en la que se penaliza a los estudiantes de acuerdo con el tiempo relativo que se toman para resolver los diferentes tipos de errores comparado con el tiempo de sus pares. En la evaluación del desempeño se demostró que el uso de este algoritmo como predictor del desempeño de los estudiantes superó algoritmos anteriores (Jadud EQ). Por otra parte, en Romero et al. (2013) se realiza la predicción del rendimiento académico de los estudiantes a partir de los foros de discusión para lo cual se utilizan tres tipos de medidas como son: cuantitativas de información estadística (número de mensajes leídos y posteados, tiempo invertido en la plataforma, entre otros); cualitativas como la calificación del contenido de los mensajes y de redes sociales (relación entre los estudiantes). En este estudio se aplican distintas técnicas de clasificación y clasificación vía agrupamiento debido a la necesidad de interpretar el modelo de mejor manera y tener una idea para la toma de decisiones, de las cuales el algoritmo EM clustering demostró tener una efectividad similar a las técnicas clásicas de clasificación basada en las medidas de precisión y F-measure.

Con base en las teorías neo piagetiana en Rubio (2020) se propone inferir las etapas de aprendizaje a través de preguntas de programación diseñadas para evaluar dichas etapas de desarrollo cognitivo. Se utiliza la técnica de k-medios. El estudio identifica a través de las calificaciones de los estudiantes tres grupos que se correspondían con el modelo. El uso de instrumentos para valorar factores intrínsecos al estudiante como la personalidad y la actitud pueden apoyar el proceso de enseñanza y aprendizaje. Ishizue et al. (2018) presentan un estudio que propone predecir la ubicación y el ranking de habilidades en un concurso de programación, para lo que utilizan atributos como la actitud, escalas psicológicas y métricas de programación, para lo cual propone tres elementos: una tarea de programación, un cuestionario a los estudiantes que resulta en una autoevaluación acerca de sus actitudes y entendimiento en clases, y una autoevaluación de escala psicológica con base en su experiencia fuera de clases. El modelo de clasificación fue desarrollado con la técnica de Support Vector Machine y tuvo una medida F de 0.912.

#### 4.1.3. Evaluación

Debido a las características de los cursos de programación, con gran cantidad de alumnos, la revisión de tareas de programación requiere de mucho tiempo del profesor, que podría utilizar en atender de manera personalizada a los alumnos con problemas. El estudio de Bai et al. (2017) realiza la evaluación de los envíos a través de una plataforma denominada trustie, la que realiza la calificación automática y el análisis de envíos; para esto, considera variables como tiempo entre envíos, número de envíos retrasados y envíos no completados. Además, analizó los errores en tareas y se pudo observar que el número de errores lógicos fue

superior a los fallos de compilación; esto permitió a los profesores dar relevancia al entrenamiento en lógica y de esta manera tomar correctivos en el curso.

Con un enfoque de aseguramiento de la calidad, Sun et al. (2019) proponen la realización de una evaluación formativa basada en la revisión por pares de las tareas de programación. Inicialmente se clasifican a los estudiantes de grupos de programación de acuerdo a su rendimiento. La calificación tiene dos componentes: los errores de programación que recibe como autor y los errores que reporta como revisor. Se plantea una estrategia escalonada de calificación. Los estudiantes mejor calificados revisan los trabajos de aquellos con menores calificaciones, mientras que el profesor revisa los trabajos de los mejores calificados. Se utiliza la técnica de k-means para determinar los grupos. En Matetic (2019) se realiza la predicción de éxito de los estudiantes a través de la visualización adicional de clases en video. Para la predicción se utiliza una red neuronal artificial y se complementa con el algoritmo de Garson y el método de perfilamiento de Lek para explicar la importancia de los predictores.

El estudio señala que la actividad de mirar videos de clases adiciones es significativa y su análisis permite identificar a estudiantes que necesitan ayuda. En el documento de Pineda y Cadavid (2019) se propone el uso de un juez en línea para la calificación automática de tareas de programación y ayuda personalizada de acuerdo con la calificación obtenida, y poder realizar el seguimiento de las retroalimentaciones realizadas por el juez y obtener patrones de comportamiento de los estudiantes. El estudio menciona que el hallazgo más importante es que la mayoría de los estudiantes tiene un comportamiento sistemático de análisis de material de estudio – intento – retroalimentación - acción correctiva de ser necesaria. El estudio de Gil, Fernández-Alemán et al. (2018) señala que el trabajo colaborativo, la evaluación continua y la calificación automática son claves para generar motivación y entusiasmo en los estudiantes. Desde ese enfoque el estudio presenta el concepto de software verde, como un uso adecuado de los recursos en el desarrollo de software; para esto se basan en los datos provistos por un sistema de juzgamiento en línea (calificación automática) que evalúa los envíos del proyecto de programación realizado por los estudiantes, de los que recupera datos específicos como fecha y hora del envío, numero del ejercicio, memoria y tiempo consumido por el programa y los resultados del programa, si es aceptado o no. Esta información se complementa con calificaciones de los estudiantes en las actividades del curso, datos de género del estudiante si el proyecto se desarrolla individualmente o en grupos de dos estudiantes. Con esta información se puede realizar la predicción de la exactitud del programa y las notas de las prácticas de los estudiantes. Se identifican dos factores asociados a software verde que son: el uso de memoria y el tiempo de ejecución.

#### 4.1.4. Retroalimentación

Así como la calificación automática puede ayudar tanto a profesores como a los estudiantes la retroalimentación también puede ser efectiva. En Gramoli, Charleston et al. (2016) se utiliza la herramienta “PASTA” (programming assessment submission and testing application), en la que los estudiantes realizan el envío de tareas de programación y la herramienta automáticamente intenta compilar el envío, si fue exitoso emite un mensaje en caso contrario le remite una retroalimentación apropiada. Esta plataforma evalúa el desempeño general del estudiante en los ejercicios, el tiempo en que la tarea fue enviada antes de la fecha de finalización de entrega, y valora el esfuerzo requerido para entregar la tarea a través del número de líneas de código, la diferencia entre el primer envío y el envío final. La retroalimentación es formativa dado que la información dada trata de modificar el comportamiento del estudiante de manera que mejore su aprendizaje, a través de la identificación de los errores lo que tiene efectos positivos en su motivación. Ott, Robins et al. (2015) realizan la retroalimentación a través de indicadores de desempeño que informan al estudiante su rendimiento en las distintas etapas, por medio de una representación gráfica y la proyección de la nota final, esto busca provocar un cambio en su comportamiento y los resultados de aprendizaje. En este estudio se señala que a pesar del uso de la información no se puede apreciar un impacto en la nota final.

## 4.2. Análisis del estudiante



### 4.2.1. Análisis de comportamiento

En su estudio, Shih (2017) utiliza una minería de patrones secuenciales a través de una herramienta de programación visual en la que deben resolver juegos a través de la programación y determina que existen diferencias de patrones de comportamiento en la interacción con la herramienta entre los estudiantes de alto desempeño y aquellos que tienen bajo desempeño. Shih (2018) utiliza una herramienta de programación visual para obtener patrones de comportamiento que permitan identificar estudiantes con alto y bajo aprovechamiento académico, en el contexto del soporte al proceso cognitivo. El estudio utiliza el enfoque de minería de patrones secuenciales. En Su, Ding y Lai (2017) se propone el análisis del comportamiento en el aprendizaje de los estudiantes a través de Facebook y determinar si existe una relación entre el nivel de participación y su rendimiento académico. A través de los registros de mensajes, acceso a materiales e interacción en la plataforma de Facebook adecuada para el curso y los ejercicios de programación realizados en otra plataforma se pueden obtener patrones de comportamiento, los cuales se visualizan y monitorean a lo largo del curso. Se pudo determinar que existió un efecto significativo en la participación de los estudiantes con relación al desempeño final de los estudiantes.

En Ayub, Toba et al. (2017) se propone identificar patrones de comportamiento y relacionarlos con el desempeño para obtener recomendación para la elaboración de un método de gamificación. Los datos de actividad son obtenidos a través de una plataforma LMS y son: el momento de acceso (mañana, tarde o noche), si accede al material durante la sesión presencial, la acción que ejecuta (leer material, realizar un ejercicio, cuestionario), también se utiliza el promedio acumulado de la carrera y la calificación final de la asignatura. El estudio demuestra que existe una fuerte correlación entre el tiempo de acceso, las actividades realizadas y la nota final. La técnica utilizada fue association rules. El uso de videos en la enseñanza de programación ha adquirido una especial relevancia debido a que libera en mucho el tiempo de los profesores y les permite enfocarse en prestar atención a los estudiantes con mayores dificultades de aprendizaje. El trabajo de Gowan, Hanna, Greer y Busch (2018) busca evaluar el impacto de clases grabadas en video, a través del análisis de patrones de visualización tratando de cuantificar y cualificar como los estudiantes se comprometen y aprenden con dichos videos y de esta manera predecir su rendimiento académico. Las variables que utiliza son la repetición de visualizaciones, la atención y el tiempo que destinan para ver el video en la clase presencial. El análisis se realizó a través de una Regresión Lineal con un  $R^2$  de 0.295.

El documento Hrženjak, Matetić y Bakarić (2015) realiza un análisis del comportamiento del uso de videos en curso de programación a través de datos obtenidos en la plataforma de LMS, y se asocia al rendimiento académico, se utiliza la técnica de association rules. Se concluye que el uso de las clases en video tiene un efecto positivo en la calidad del curso y muestran conexión con las notas finales. El estudio presentado en Amira, Lamia y Mohamed (2018) señala que el uso de plataformas en línea trae como beneficio un aprendizaje adaptativo, de esta manera se puede aprovechar más esta característica si identificamos el estilo de aprendizaje que tienen los estudiantes y ajustamos el contenido que se presenta a ellos, esto redundará en su satisfacción subjetiva y la velocidad y desempeño en el aprendizaje. Para este estudio se aplicó el modelo de estilos de aprendizaje Felder-Silverman y la metodología de aprendizaje basado en problemas. Los problemas fueron presentados en formatos de imagen, video y texto. El contenido fue personalizado de acuerdo con la evaluación de criterios de nivel y estilo identificados por el modelo.

### 4.2.2. Visualización

En el estudio realizado en Matsuzawa, Tanaka, Kitani y Sakai (2017) se propone una representación gráfica (dashboard) para que el profesor pueda interpretar y mejorar el proceso de enseñanza aprendizaje de la clase que imparte, a partir del nivel de participación de los estudiantes y métricas sobre los ejercicios de programación como son: el tiempo de trabajo, líneas de código y tiempos de corrección de errores de compilación, de esta manera el profesor puede medir el avance de los estudiantes y tomar acción en determinado momento. Los autores Matetić, Bakarić y Sisović (2015) realizan el análisis de comportamiento a través de visualización con la herramienta weka, esto permite relacionar el uso de materiales como uso de videos de clases, autoevaluaciones y cuestionarios, con la nota final determinada como “bien”, “aceptable” y

“fracaso”. Luego se aplica la técnica de association rules y se obtienen resultados significativos sobre el uso de materiales, en particular los videos de clases y las autoevaluaciones.

Para monitorear efectivamente el trabajo en equipo y dar una retroalimentación significativa hacia los estudiantes, en Tardmazdi et al. (2015) se propone la creación de un cuadro de mando (dashboard), que con base en los datos de discusiones en línea permite revisar el estado de cada equipo de a través de un gráfico de tipo araña sobre el desempeño de los roles de trabajo en equipo en el cual se visualizan las principales variables. La evolución de las emociones también se analiza mediante el análisis de sentimientos en la colaboración del equipo. En Leony, Muñoz-Merino, Pardo y Kloos (2013) también se presenta una herramienta de visualización de las emociones durante todo el curso, las emociones son alegría, confusión, frustración y aburrimiento.

El estudio muestra como los profesores pueden visualizar e interpretar la conexión entre los resultados emocionales y los de aprendizaje. El estudio plantea la inferencia de las emociones a través de datos de bajo nivel, obtenidos de las interacciones de los estudiantes con la plataforma, es así como, un estudiante confundido es aquel que realiza un estilo de programación de prueba y error; y, un estudiante aburrido será aquel que realiza búsquedas que no tiene relación con el curso. Las visualizaciones se clasifican basadas en el tiempo, en el contexto, los cambios emocionales y la información acumulada. En Gampa y Baynes (2019) se reporta el desarrollo de una herramienta visual de analítica que permita el monitoreo del progreso de los estudiantes e identificar, por ejemplo, qué parte de la tarea de programación provoca más errores o qué pregunta de un examen es la mejor para evaluar el análisis de corridas. Los datos analizados contienen información las actividades realizadas durante el desarrollo de seis tópicos de clase.

## 4.3. Recursos

### 4.3.1. Herramientas

En Hunter, Livingstone, Neve y Alsop (2013) se plantea la necesidad de integrar ambientes para asistir el proceso de enseñanza aprendizaje en programación. Se detalla la implementación de la herramienta noolab un ambiente de enseñanza aprendizaje para distintos cursos de programación. El sistema provee un ambiente donde se entregan los contenidos tanto de principios algorítmicos y sintaxis, así como para la edición y prueba de código. Proporciona también una retroalimentación derivada no solo de la respuesta a los problemas de programación sino también con otras interacciones con el ambiente de aprendizaje. También incorpora mecanismos de gamificación como la entrega de medallas que han tenido un impacto positivo en la motivación y participación de los estudiantes.

El análisis cualitativo de patrones provenientes de las interacciones de los estudiantes con el sistema pudo identificar deficiencias en los materiales de aprendizaje, así como un potencial plagio. El análisis cuantitativo pudo identificar una correlación entre las corridas de los programas y la nota final. El estudio de Fu, Shimada et al. (2017) plantea el desarrollo de un plug-in para moodle denominado Learning Analytics in Programming Language Education (LAPLE), que tiene una representación gráfica sobre el comportamiento de los estudiantes en la clase e identifica las dificultades que afrontan en determinada situación de aprendizaje lo que permite al profesor tomar decisiones sobre la mejora de materiales en tiempo real. Se identifican las páginas más visitadas por los estudiantes para sugerir soluciones a los problemas que tienen los estudiantes en determinado momento, esto ayuda a los profesores a entender qué materiales requieren especial atención, análisis y mejora. En Guerrero-Higuera et al. (2018) se presenta la herramienta Model Evaluator (MoEv) que permite evaluar distintos algoritmos con el objetivo de elegir el mejor modelo predictivo para clasificar estudiantes que finalizarán la tarea de programación y aquellos que no.

El enfoque de este estudio parte el monitoreo y la interacción con sistema de control de versiones, se evalúa la actividad de los estudiantes en la plataforma GIT, el número de operaciones commit, número de días entre cada operación, promedio de operaciones en el día, número de líneas adicionadas durante una sesión,



entre otros. En Guerrero-Higueras, Matellán-Olivera et al. (2018) se amplía este estudio con la inclusión de una prueba de autor en la que se pide modificar un segmento de código. Pereira (2016) realizó el diseño e implementación de un chatbot @dawebot para la práctica de exámenes, los estudiantes pueden elegir la prueba que quieren practicar, el bot presenta una pregunta y el estudiante responde, en ese momento el bot presenta una retroalimentación.

El estudio presentado por Azcona, Hsiao y Smeaton (2019) implementa una plataforma denominada PredictCS, que detecta automáticamente los estudiantes en riesgo de fallar en el siguiente examen de laboratorio, para lo cual usa los datos de historia académica, el envío de tareas de programación y los registros de la plataforma, utilizando la técnica de K-means. Otro posible abordaje de la elaboración de herramientas es la gamificación, que a través de los instrumentos lúdicos permiten una mejora en el desempeño del estudiante. Los juegos de simulación también ofrecen experiencias en ambientes de aprendizaje controlados y sin riesgos. El estudio propuesto por Larenas, Marín y Giachetti (2018) describe el desarrollo de una herramienta denominada lassutopia, concebida como un juego serio dedicado al apoyo a la enseñanza modelo conceptual en ingeniería de software a través de la construcción de un robot y sucesivas misiones. Al finalizar el estudiante responde un cuestionario para obtener la percepción de la usabilidad del juego como nueva técnica en el proceso de enseñanza aprendizaje. Los resultados mostraron que todos los estudiantes lograron aprender, a través del juego, como construir un robot y detectar los defectos de atributos de clase en un nivel básico. La mayoría de los estudiantes identificaron los defectos en niveles medios y altos.

En Ayub, Tobaet et al. (2019) se presenta un estudio en el que desarrolla los cuestionarios en línea a manera de torneo, el resultado actualiza una tabla de clasificación del curso, donde los estudiantes pueden ver su desempeño. Se valora el aporte de esta modalidad de cuestionarios al proceso de aprendizaje aplicando una técnica de association rules y para validar la robustez de las reglas generadas se aplica la técnica de clasificación "Decision Trees". El estudio realizado en Laakso, Kaila y Rajala (2018) describe el diseño e implementación de una herramienta denominada "ViLLE" que permite la colaboración entre profesores, estudiantes, la evaluación y la retroalimentación automática y la visualización del proceso. El enfoque de desarrollo se basa en el compromiso del estudiante con el proceso de aprendizaje a través de la visualización del proceso, a mayor compromiso mejores resultados de aprendizaje. El estudio señala que la filosofía detrás de la herramienta pretende ahorrar recursos de los profesores para que puedan redirigirlos en formas más productivas de enseñanza. La herramienta presentada en Petit, Roura et al. (2018) se denomina judge.org, es un evaluador de tareas de programación en línea y está diseñada para el beneficio tanto de alumnos como de profesores, siguiendo un enfoque de aprender haciendo. La solución consiste en realizar una minería de datos sobre las entregas de tareas pasadas y de manera particular sobre sus errores y generar casos de prueba que capturen la problemática; y, a partir de esto generar la recomendación apropiada para cada estudiante que ha enviado su tarea.

## 4.4. Teóricos

### 4.4.1. Marcos de trabajo

Para evaluar de manera correcta la efectividad de los foros de discusión, en Yoo y Kim (2014) se presenta un marco de trabajo para el uso de la participación de los estudiantes en discusiones en línea como predictor del desempeño de la clase de proyectos de ingeniería de sistemas. A través del análisis cualitativo y cuantitativo se establece la relación entre variables como el grado de participación o cantidad de mensajes posteados, el tipo de palabras que usan o la coherencia en los mensajes, la expresión de emociones (denotan una actitud positiva o negativa), el tipo de participación; es decir, el rol de un estudiante si busca o proporciona información y la periodicidad del trabajo. En Mohamad, Mumtazimah y Fadzli (2013) se propone un marco de trabajo para una enseñanza personalizada en línea para lo cual toma la teoría de ítem respuesta y felder-silverman para estimar su habilidad y sugerir los materiales más apropiados para cada estudiante. La dificultad del material se ajusta dinámicamente usando una red neuronal artificial en la que se evalúa el nivel de entendimiento de los estudiantes. El aprendizaje personalizado busca proveer a cada estudiante el contenido justo y necesario

conforme su propio nivel de conocimiento, comportamiento y perfil.

#### 4.4.2. Guías, estrategias e instrumentos

En Moretti, González-Brenes y McKnight (2014) se propone una metodología que evalúa la calidad de los profesores a través de opiniones de los estudiantes y la disponibilidad del sílabo en línea, la elección de la clase y la correlación con los resultados académicos. El estudio presentado en Azcona, Hsiao, Arora y Smeaton (2019) propone el desarrollo de una metodología que permita determinar el perfil individual de los estudiantes basado en el código de programación, para lo cual investiga sobre la forma de representación del código del estudiante en un formato vectorial, esta representación permite comparar efectivamente a los estudiantes y establecer grupos con comportamientos similares y realizar predicciones sobre estos. Dada la naturaleza de la programación el plagio se ha convertido en un importante problema que requiere atención. Ullah, Wang et al. (2018) proponen una metodología para identificación del plagio a través del cálculo de la similitud semántica entre tareas de programación, para lo cual se utiliza el procesamiento natural del lenguaje y de manera particular la técnica de Análisis Semántico Latente. Mediante este estudio se logró hallar similitudes semánticas entre varios ejercicios analizados en lenguajes de C++ y Java. El incremento del uso de dispositivos móviles también ha generado una oportunidad en el ámbito de la academia.

En Matthew, Adepoju et al. (2018) se detalla el diseño e implementación de una interfaz móvil para el trabajo con modelos predictivos que permitan la mejora del desempeño de los estudiantes. El objetivo de esta aplicación es tratar de identificar los factores ocultos que son significantes en el desempeño académico de los estudiantes. Para esto se diseñó un cuestionario que analiza 21 factores y 81 variables. Para la construcción de los modelos predictivos se utilizó las técnicas de regresión lineal y árboles de decisión. Como resultado se obtuvo que la actitud de estudiantes y profesores, la percepción de miedo de los estudiantes, el flujo irregular de energía eléctrica, las facilidades de la universidad, la salud de los estudiantes y la asistencia de los estudiantes son significantes al desempeño académico. La publicación de Krauss, Merceron et al. (2017) describe una aplicación móvil para el acompañamiento del aprendizaje, la que ofrece materiales de clase como textos, videos, códigos fuente o animaciones. La aplicación analiza las interacciones de los estudiantes identifica debilidades en el aprendizaje y propone recomendaciones para mejorar el rendimiento académico. El estudio pudo tres grupos de estudiantes de acuerdo a sus notas finales. Uno de los hallazgos realizados fue que el 40% de las interacciones provenían de dispositivos móviles, lo cual se puede explicar por el hecho que los estudiantes prefieren en el mismo ambiente en el que programan.

### 4.5. Análisis del profesor

#### 4.5.1. Materiales usados

El uso de estrategias, actividades y materiales puede ser determinante en la motivación del estudiante por lo que su análisis puede servir para mejorar el proceso de enseñanza aprendizaje. Precisamente el estudio de Ruiz-Ferrández, Ortega y Roca-Piera (2018) analiza el uso de diferentes elementos formativos para incrementar el interés de los estudiantes. Desde la perspectiva del “aprendizaje activo” analiza el uso de materiales que involucren la participación comprometida de los estudiantes como pueden ser los foros de discusión, la solución de problemas, los estudios de caso, los juegos de roles, entre otros. En este estudio se destaca la participación del profesor como guía de este proceso. En McGowan, Hanna y Anderson (2016) se analiza el uso de clases impartidas por videos. Este experimento se realizó a través de un canal cerrado youtube, donde solo pueden participar los estudiantes invitados. La plataforma provee la analítica de datos. En este estudio se pudo analizar algunas variables como la tasa de abandono y la retención de audiencia. Una conclusión del estudio es que la retención de audiencia desciende mientras más largo es el video, a partir de esto presenta algunas recomendaciones para el diseño de videos como, videos más largos son menos atractivos, el tiempo de duración puede ser de máximo diez minutos, se valoran más las demostraciones prácticas de código relacionado con la teoría más que la teoría sola y la presentación de láminas con código cautiva más que una lámina solo con texto. En el estudio de Priss (2013) se analiza desde la perspectiva del formal concept analysis para identificar dificultades conceptuales que tienen los estudiantes con los ejercicios de programación,



con base en una herramienta de evaluación. La aplicación de esta técnica permitió depurar la información para un análisis posterior.

### 4.5.2. Currículo

Wang (2016) realiza un análisis de redes de aprendizaje para un curso de programación a través de un análisis de clúster. El curso tiene doce puntos de conocimiento (temas de programación en C); primero, realiza un agrupamiento en cada grupo se tiene un promedio de rendimiento; luego, analiza los puntos de conocimiento con cada grupo y puede determinar en qué temas son fuertes o débiles cada grupo. Se utiliza la técnica de k-means.

### 4.5.3. Mejora de materiales

A partir de la proliferación de los sistemas para la Gestión del Aprendizaje, los datos de los estudiantes en el proceso de enseñanza aprendizaje son capturados de manera permanente, lo que permite analizar la relación entre estos y los materiales utilizados en el proceso y proponer mejoras en los métodos de enseñanza. Los autores Chu, Yu, Jiang y Wang (2018) investigan el diseño educacional de recursos en ambientes de aprendizaje apoyados con tecnología también conocidos como Blended Learning o Technology Enhanced Learning. Para esto el estudio realiza una encuesta inicial a los estudiantes para diseñar las actividades para los momentos presenciales y para aquellos que requieran un estudio en línea. Se aplican encuestas sucesivas de manera que se permite adaptar los materiales dinámicamente. En este caso se señala que el uso de video en actividades no presenciales con el enfoque de clase invertida, permiten una mayor participación de los estudiantes y un mejor aprovechamiento del tiempo del profesor en la interacción con los estudiantes. En Hao, Galyardt, Barnes, Branch y Wright (2019) se realiza una identificación automática de preguntas ineficaces para el aprendizaje que permita alertar a los estudiantes para que revisen sus preguntas y sugerirles preguntas más adecuadas, para esto se recolectó las preguntas de semestres anteriores y se procedió a una clasificación manual con expertos humanos identificando la relevancia para el aprendizaje y su eficacia como pregunta. Para la clasificación de cada nueva pregunta se utilizó varios algoritmos de aprendizaje de máquinas. El estudio reporta que los resultados de clasificación estaban lejos de ser satisfactorios con un 57.8% de precisión.

RO2 ¿Cuáles son los potenciales temas de investigación en la integración de analítica de la enseñanza y el aprendizaje para cursos de programación en educación superior?

Uno de los temas de investigación que se desprende de esta revisión es que todavía existen factores ocultos (Matthew, Adepoju et al, 2018) que tienen influencia sobre el rendimiento académico y que tienen relación con los estudiantes, los profesores, los ambientes de aprendizaje y la familia, así como la interacción entre ellos no ha sido estudiada exhaustivamente.

Las teorías educativas señalan que el aprendizaje activo es necesario para lograr el involucramiento y compromiso de los estudiantes con el proceso de enseñanza aprendizaje según señalan Ruiz-Ferrández, Ortega y Roca-Piera (2018); en este sentido la motivación juega un rol preponderante en el diseño y la aplicación de estrategias y metodologías de aprendizaje para alcanzar una mejora en el desempeño de los estudiantes de programación. La combinación de características estáticas y dinámicas de los estudiantes también requieren de un análisis mayor (Azcona, Hsiao & Smeaton, 2019).

El diseño curricular en cursos de programación un tema tiene posibilidades de investigación Hao, Galyardt et al. (2019), si se considera de los estudiantes y profesores, es interesante el aporte que puede dar el TLA al diseño instruccional.

Son muchos los estudios que se enfocan en el soporte a los estudiantes y pocos aquellos que proponen el soporte al profesor, documentado por Ishizue, Sakamoto, Washizaki y Fukazawa (2018). Para los autores Guerrero-Higueras, DeCastro-García, Conde y Matellán (2018) el potencial del TLA para entender y optimizar el proceso de enseñanza aprendizaje y los entornos en los que ocurren. Desde el enfoque de la

integración del TLA no existe un estudio que realice esta integración y que pueda concretar ese potencial en el ámbito de la educación en programación.

De otra parte, el estudio de Knight y Buckingham (2017) señala que es necesario reflexionar sobre los enfoques epistemológicos, pedagógicos y de evaluación que dan relevancia a qué es lo que nos interesa medir, en términos de conocimiento, cómo lo medimos y por qué ese conocimiento es importante para nuestra investigación.

De acuerdo con la publicación de Ullah, Wang et al. (2018) la ética en TLA también es un tema importante que requiere un mayor estudio, ya sea vista desde la óptica del plagio o fraude, así como también en el uso de datos del proceso de enseñanza y aprendizaje, la propiedad de esos datos, sus implicaciones de política educativa también requieren de un mayor estudio. Debido a la realidad que impone el cambio climático, es necesario investigar los posibles impactos ambientales y su mitigación (Gil, Fernández-Alemán, Trujillo, García-Mateos, Luján-Mora & Toval, 2018).

## 5. Conclusiones

El proceso de enseñanza aprendizaje de programación en cursos iniciales de educación superior es un problema de investigación vigente que capta la atención de investigadores en el mundo y que no solo se ubica en el área de las ciencias de la computación, sino en distintos dominios de las ciencias, dado que la programación se entiende como una nueva competencia necesaria en el contexto actual de la sociedad.

El uso de la analítica como propuesta de aplicación en el proceso de enseñanza aprendizaje es amplio. Las posibilidades de implementación, como se ha podido notar en la revisión, son extensas y dependen del enfoque que se quiera aplicar. Los estudios proponen resultados disímiles, algunos con más éxito que otros. Esto implica que es necesario continuar con las investigaciones en este tema.

Se ha podido identificar oportunidades de investigación en la aplicación de Analítica de Enseñanza y Aprendizaje en cursos de programación para educación superior, ya sea en el desarrollo de modelos de predicción de rendimiento, soporte a alumnos y profesores, diseño curricular, ética o políticas educativas. Sin embargo, es importante señalar que, a pesar de que, algunos estudios son similares en cuanto a su objetivo, no se puede apreciar una metodología común para abordar estos problemas.

La disponibilidad de datos masivos en el ámbito educativo abre la oportunidad de aplicación de la analítica para mejorar la calidad del proceso de enseñanza aprendizaje, por lo que es necesario capitalizar esta oportunidad en beneficio de la sociedad y los profesionales que integren la programación como una competencia.

Cómo citar este artículo / How to cite this paper

Fiallos Quinteros, J. C.; Jiménez Builes, J. A.; Branch Bedoya, J. W. (2022). Analítica de enseñanza y aprendizaje en cursos de programación. *Campus Virtuales*, 11(1), 35-49.  
<https://doi.org/10.54988/cv.2022.1.880>

## Referencias

- Ahadi, A. (2016). Early identification of novice programmers' challenges in coding using machine learning techniques. In ICER 2016 - Proceedings of the 2016 ACM Conference on International Computing Education Research (pp. 263-264). doi:10.1145/2960310.2960339.
- Amira, T.; Lamia, M.; Mohamed, H. (2018). A collaborative algorithmic problem-based learning environment using learners' learning



- styles. In ACM International Conference Proceeding Series. doi:10.1145/3227696.3227724.
- Ayub, M.; Toba, H.; Wijanto, M. C.; Yong, S.; Wijaya, B. (2019). Gamification for blended learning in higher education. *World Transactions on Engineering and Technology Education*, 17(1), 76-81.
- Ayub, M.; Toba, H.; Yong, S.; Wijanto, M. C. (2017). Modelling students' activities in programming subjects through educational data mining. *Global Journal of Engineering Education*, 19(3), 249-255.
- Azcona, D.; Hsiao, I.-H.; Smeaton, A. F. (2019). Detecting students-at-risk in computer programming classes with learning analytics from students' digital footprints. *User Modeling and User-Adapted Interaction*, 29(4), 759-788. doi:10.1007/s11257-019-09234-7.
- Azcona, D.; Hsiao, I.-H.; Smeaton, A. F. (2019). Personalizing computer science education by leveraging multimodal learning analytics. In *Proceedings - Frontiers in Education Conference, FIE*. doi:10.1109/FIE.2018.8658596.
- Azcona, D.; Hsiao, I.-H.; Arora, P.; Smeaton, A. (2019). User2Code2vec: Embeddings for profiling students based on distributional representations of source code. In ACM International Conference Proceeding Series (pp. 86-95). doi:10.1145/3303772.3303813.
- Azcona, D.; Smeaton, A. F. (2017). Targeting at-risk students using engagement and effort predictors in an introductory computer programming course. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 10474 LNCS (pp. 361-366). doi:10.1007/978-3-319-66610-5\_27.
- Badr, G.; Algobail, A.; Almutairi, H.; Almutery, M. (2016). Predicting Students' Performance in University Courses: A Case Study and Tool in KSU Mathematics Department. *Procedia Computer Science*, 82, 80-89. doi:10.1016/j.procs.2016.04.012.
- Bai, Y.; Chen, L.; Yin, G.; Mao, X.; Deng, Y.; Wang, T.; Lu, Y.; Wang, H. (2017). Quantitative analysis of learning data in a programming course. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 10179 LNCS (pp. 436-441). doi:10.1007/978-3-319-55705-2\_37.
- Casey, K.; Azcona, D. (2017). Utilizing student activity patterns to predict performance. *International Journal of Educational Technology in Higher Education*, 14(1). doi:10.1186/s41239-017-0044-3.
- Costa, E. B.; Fonseca, B.; Santana, M. A.; de Araújo, F. F.; Rego, J. (2017). Evaluating the effectiveness of educational data mining techniques for early prediction of students' academic failure in introductory programming courses. *Computers in Human Behavior*, 73, 247-256. doi:10.1016/j.chb.2017.01.047.
- Chatti, M.; Dyckhoff, A.; Schroeder, U.; Thüs, H. (2012). A Reference Model for Learning Analytics. *International Journal of Technology Enhanced Learning (IJTEL)*.
- Chu, Q.; Yu, X.; Jiang, Y.; Wang, H. (2018). Data analysis of blended learning in python programming. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 11336 LNCS (pp. 209-217). doi:10.1007/978-3-030-05057-3\_16.
- Ferreira, M.; Avitabile, C.; Botero J.; Haimovich, F.; Urzúa, S. (2017). At a Crossroads Higher Education in Latin America and the Caribbean. *International Bank for Reconstruction and Development / The World Bank*. Washington, United States of America. doi:10.1596/978-1-4648-1014-5.
- Fu, X.; Shimada, A.; Ogata, H.; Taniguchi, Y.; Suehiro, D. (2017). Real-time learning analytics for C programming language courses. In ACM International Conference Proceeding Series (pp. 280-288). doi:10.1145/3027385.3027407.
- Gampa, R.; Baynes, A. (2019). Visual analytic workflow to understand students' performance in computer science courses. In *Proceedings - Frontiers in Education Conference, FIE*. doi:10.1109/FIE.2018.8658790.
- Garbanzo, G. (2007). Factores asociados al rendimiento académico en estudiantes universitarios, una reflexión desde la calidad de la educación superior pública. *Revista Educación*, 31(1), 43-63.
- Gil, D.; Fernández-Alemán, J. L.; Trujillo, J.; García-Mateos, G.; Luján-Mora, S.; Toval, A. (2018). The effect of green software: A study of impact factors on the correctness of software. *Sustainability*, 10(10), 3471. doi:10.3390/su10103471.
- Gowan, A. M.; Hanna, P.; Greer, D.; Busch, J. (2018). Video makes the coding star?. In 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2018 - Proceedings (pp. 610-615). doi:10.23919/MIPRO.2018.8400115.
- Gramoli, V.; Charleston, M.; Jeffries, B.; Radu, A.; Koprinska, I.; Viglas, A.; McGrane, M.; Yacef, K. (2016). Mining autograding data in computer science education. In ACM International Conference Proceeding Series. doi:10.1145/2843043.2843070.
- Greller, W.; Drachler, H. (2012). Translating learning into numbers: A generic framework for learning analytics. *Educational Technology and Society*, 15(3), 42-57.
- Guerrero-Higueras, Á. M.; DeCastro-García, N.; Conde, M. Á.; Matellán, V. (2018) Predictive models of academic success: A case study with version control systems. In ACM International Conference Proceeding Series (pp. 306-312). doi:10.1145/3284179.3284235.
- Guerrero-Higueras, Á. M.; Matellán-Olivera, V.; Costales, G. E.; Fernández-Llamas, C.; Rodríguez-Sedano, F. J.; Conde, M. Á. (2018). Model for evaluating student performance through their interaction with version control systems. *CEUR Workshop Proceedings*, 2188, 104-112.
- Hao, Q.; Galyardt, A.; Barnes, B.; Branch, R. M.; Wright, E. (2019). Automatic identification of ineffective online student questions in computing education. In *Proceedings - Frontiers in Education Conference, FIE*. doi:10.1109/FIE.2018.8658642.
- Hrženjak, M. P.; Matetić, M.; Bakarić, M. B. (2015). Mining association rules in learning management systems. In 2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2015 - Proceedings. doi:10.1109/MIPRO.2015.7160418.
- Hunter, G.; Livingstone, D.; Neve, P.; Alsop, G. (2013). Learn Programming++: The Design, Implementation and Deployment of an Intelligent Environment for the Teaching and Learning of Computer Programming. In *Proceedings - 9th International Conference on Intelligent Environments, IE 2013*, art. no. 6597801 (pp. 129-136). doi:10.1109/IE.2013.46.
- Infante-Moro, A.; Infante-Moro, J. C.; Gallardo-Pérez, J. (2020a). Key factors in the implementation of Cloud Computing as a service and communication tool in universities. In *Eighth International Conference on Technological Ecosystems for Enhancing Multiculturality*

- (pp. 631-636). New York, NY, USA: ACM International Conference Proceeding Series. doi:10.1145/3434780.3436698.
- Infante-Moro, A.; Infante-Moro, J. C.; Gallardo-Pérez, J. (2020b). Las posibilidades de empleo del Internet de las Cosas en el sector hotelero y sus necesidades formativas. *Education in the knowledge society*, (21), 14. doi:10.14201/eks.22777.
- Infante-Moro, A.; Infante-Moro, J. C.; Gallardo-Pérez, J. (2021a). Análisis de las competencias digitales en el Máster de Turismo de la Universidad de Huelva. *Campus Virtuales*, 10(2), 141-151.
- Infante-Moro, A.; Infante-Moro, J. C.; Gallardo-Pérez, J.; Luque-de la Rosa, A. (2021b). Motivational Factors in the Use of Videoconferences to Carry out Tutorials in Spanish Universities in the Post-Pandemic Period. *International Journal of Environmental Research and Public Health*, 18(19), 10474. doi:10.3390/ijerph181910474.
- Ishizue, R.; Sakamoto, K.; Washizaki, H.; Fukazawa, Y. (2018). Student placement and skill ranking predictors for programming classes using class attitude, psychological scales, and code metrics. *Research and Practice in Technology Enhanced Learning*, 13(1), 7. doi:10.1186/s41039-018-0075-y.
- Knight, S.; Buckingham Shum, S. (2017). Theory and Learning Analytics. In C. Lang, G. Siemens, A. F. Wise, & D. Gaevic (Eds.), *The Handbook of Learning Analytics* (pp. 17-22).
- Knight, S.; Shum, S. B.; Littleton, K. (2014). Epistemology, Assessment, Pedagogy: Where Learning Meets Analytics in the Middle Space. *Journal of Learning Analytics*, 1(2), 23-47.
- Krauss, C.; Merceron, A.; An, T.-S.; Zwicklbauer, M.; Steglich, S.; Arbanowski, S. (2017). Teaching advanced web technologies with a mobile learning companion application. In *ACM International Conference Proceeding Series*. doi:10.1145/3136907.3136937.
- Laakso, M.-J.; Kaila, E.; Rajala, T. (2018). ViLLE – collaborative education tool: Designing and utilizing an exercise-based learning environment. *Education and Information Technologies*, 23(4), 1655-1676. doi:10.1007/s10639-017-9659-1.
- Lang, C.; Siemens, G.; Wise, A. F.; Gaevic, D. (2017). *The Handbook of Learning Analytics*. Society for Learning Analytics Research (SoLAR), Alberta, Canada, 1 edition.
- Larenas, F.; Marín, B.; Giachetti, G. (2018). Classutopia: A serious game for conceptual modeling design. In *Proceedings of the International Conference on Software Engineering and Knowledge Engineering, SEKE* (pp. 116-121). doi:10.18293/SEKE2018-145.
- Leony, D.; Muñoz-Merino, P. J.; Pardo, A.; Kloos, C. D. (2013). Provision of awareness of learners' emotions through visualizations in a computer interaction-based environment. *Expert Systems with Applications*, 40(13), 5093-5100. doi:10.1016/j.eswa.2013.03.030.
- Matetic, M. (2019). Mining learning management system data using interpretable neural networks. In *2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2019 – Proceedings* (pp. 1282-1287). doi:10.23919/MIPRO.2019.8757113.
- Matetic, M.; Bakaric, M. B.; Sisovic, S. (2015). Association rule mining and visualization of introductory programming course activities. In *ACM International Conference Proceeding Series* (pp. 374-381). doi:10.1145/2812428.2812438.
- Matthew, F. T.; Adepoju, A. I.; Ayodele, O.; Olumide, O.; Olatayo, O.; Adebimpe, E.; Bolaji, O.; Funmilola, E. (2018). Development of mobile-interfaced machine learning-based predictive models for improving students' performance in programming courses. *International Journal of Advanced Computer Science and Applications*, 9(5), 105-115. doi:10.14569/IJACSA.2018.090514.
- Matsuzawa, Y.; Tanaka, Y.; Kitani, T.; Sakai, S. (2017). A demonstration of evidence-based action research using information dashboard in introductory programming education. *IFIP Advances in Information and Communication Technology*, 515, 619-629. doi:10.1007/978-3-319-74310-3\_62.
- McGowan, A.; Hanna, P.; Anderson, N. (2016). Teaching programming - Understanding lecture capture youtube analytics. In *Annual Conference on Innovation and Technology in Computer Science Education, ITiCSE* (pp. 35-40). doi:10.1145/2899415.2899421.
- Ministerio de Educación Nacional Colombia (2012). *Educación Superior. Boletín No. 19 Calidad de la Educación Superior. El camino a la prosperidad*. Bogotá, Colombia
- Mohamad, F. S.; Mumtazimah, M.; Fadzli, S. A. (2013). Integrating an e-learning model using IRT, Felder-Silverman and neural network approach. In *2013 2nd International Conference on Informatics and Applications, ICIA 2013* (pp. 207-211). doi:10.1109/ICoIA.2013.6650257.
- Mora García, R. T. (2015). Factors involved in university academic performance: A case study. [Factores que intervienen en el rendimiento académico universitario: Un estudio de caso]. *Opcion*, 31(Special Issue 6), 1041-1063.
- Morales Salas, R. E.; Infante-Moro, J. C.; Gallardo-Pérez, J. (2019). La mediación e interacción en un AVA para la gestión eficaz en el aprendizaje virtual. *Campus Virtuales*, 8(1), 49-61.
- Moretti, A.; González-Brenes, J. P.; McKnight, K. (2014). Mining the web to leverage collective intelligence and learn student preferences. *CEUR Workshop Proceedings*, 1183, 100-106.
- Mortenson, M.; Doherty, N.; Robinson, S. (2014). Operational research from Taylorism to Terabytes: A research agenda for the analytics age. School of Business and Economics, Loughborough University, Loughborough, Leicestershire, United Kingdom.
- Ott, C.; Robins, A.; Haden, P.; Shephard, K. (2015). Illustrating performance indicators and course characteristics to support students' self-regulated learning in CS1. *Computer Science Education*, 25(2), 174-198. doi:10.1080/08993408.2015.1033129.
- Peña-Ayala, A. (2018). Learning analytics: A glance of evolution, status, and trends according to a proposed taxonomy. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8(3). doi:10.1002/widm.1243.
- Pereira, J. (2016). Leveraging chatbots to improve self-guided learning through conversational quizzes. In *ACM International Conference Proceeding Series* (pp. 911-918). doi:10.1145/3012430.3012625.
- Petit, J.; Roura, S.; Carmona, J.; Cortadella, J.; Duch, J.; Giménez, O.; Mani, A.; Mas, J.; Rodríguez-Carbonell, E.; Rubio, E.; De San Pedro, E.; Venkataramani, D. (2018). Jutge.org: Characteristics and Experiences. *IEEE Transactions on Learning Technologies*, 11(3), 321-333. doi:10.1109/TLT.2017.2723389.
- Petkovic, D.; Sosnick-Perez, M.; Huang, S.; Todtenhoefer, R.; Okada, K.; Arora, S.; Sreenivasen, R.; Flores, L.; Dubey, S. (2015). SETAP: Software engineering teamwork assessment and prediction using machine learning. In *Proceedings - Frontiers in Education Conference*,



FIE. doi:10.1109/FIE.2014.7044199.

Pineda, A. F.; Cadavid, J. M. (2019). Traceability of learning activities in computer programming courses using an automatic online judge within an LMS [Trazabilidad de actividades de aprendizaje en cursos de programación de computadores usando un juez automático en línea dentro de un LMS]. *CEUR Workshop Proceedings*, 2425, 122-131.

Priss, U. (2013). Using FCA to analyse how students learn to program. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 7880 LNAI, 216-227. doi:10.1007/978-3-642-38317-5\_14.

Romero, C.; López, M.-I.; Luna, J.-M.; Ventura, S. (2013). Predicting students' final performance from participation in on-line discussion forums. *Computers and Education*, 68, 458-472. doi:10.1016/j.compedu.2013.06.009.

Rubio, M. A. (2020). Automatic Categorization of Introductory Programming Students. *Advances in Intelligent Systems and Computing*, 951, 302-311. doi:10.1007/978-3-030-20005-3\_31.

Ruiz-Fernández, M.; Ortega, G.; Roca-Piera, J. (2018). Learning analytics and evaluative mentoring to increase the students' performance in computer science. In *IEEE Global Engineering Education Conference, EDUCON* (pp. 1297-1304). doi:10.1109/EDUCON.2018.8363379.

Sagar, M.; Gupta, A.; Kaushal, R. (2016). Performance prediction and behavioral analysis of student programming ability. In *2016 International Conference on Advances in Computing, Communications and Informatics, ICACCI 2016* (pp. 1039-1045). doi:10.1109/ICACCI.2016.7732181.

Sergis, S.; Sampson D. G. (2017). Teaching and Learning Analytics to Support Teacher Inquiry: A Systematic Literature Review. In A. Peña-Ayala (Ed.), *Learning Analytics: Fundamentals, Applications, and Trends. Studies in Systems, Decision and Control*. Springer, Cham.

Shih, W.-C. (2018). Mining Sequential Patterns to Explore Users' Learning Behavior in a Visual Programming App. In *Proceedings - International Computer Software and Applications Conference* (pp. 126-129). doi:10.1109/COMPSAC.2018.10216.

Shih, W.-C. (2017). Mining Learners' Behavioral Sequential Patterns in a Blockly Visual Programming Educational Game. In *2017 International Conference on Industrial Engineering, Management Science and Application, ICIMSA 2017*. doi:10.1109/ICIMSA.2017.7985594.

Su, Y.-S.; Ding, T.-J.; Lai, C.-F. (2017). Analysis of students engagement and learning performance in a social community supported computer programming course. *Eurasia Journal of Mathematics, Science and Technology Education*, 13(9), 6189-6201. doi:10.12973/eurasia.2017.01058a.

Sun, Q.; Wu, J.; Rong, W.; Liu, W. (2019). Formative assessment of programming language learning based on peer code review: Implementation and experience report. *Tsinghua Science and Technology*, 24(4), 423-434. doi:10.26599/TST.2018.9010109.

Tama, B. A.; Rhee, K.-H. (2018). A comparative study of classifier ensembles for detecting inactive learner in university. *International Journal of Data Analysis Techniques and Strategies*, 10(4), 351-368. doi:10.1504/IJDATS.2018.095216.

Tarmazdi, H.; Vivian, R.; Szabo, C.; Falkner, K.; Falkner, N. (2015). Using learning analytics to visualise computer science teamwork. In *Annual Conference on Innovation and Technology in Computer Science Education, ITICSE* (pp. 165-170). doi:10.1145/2729094.2742613.

Ullah, F.; Wang, J.; Farhan, M.; Jabbar, S.; Wu, Z.; Khalid, S. (2018). Plagiarism detection in students' programming assignments based on semantics: multimedia e-learning based smart assessment methodology. *Multimedia Tools and Applications*. doi:10.1007/s11042-018-5827-6.

Wang, L. (2016) Network teaching system based on a clustering analysis algorithm. *World Transactions on Engineering and Technology Education*, 14(1), 179-183.

Watson, C.; Li, F. W. B. (2014). Failure rates in introductory programming revisited. In *ITICSE 2014 - Proceedings of the 2014 Innovation and Technology in Computer Science Education Conference* (pp. 39-44). doi:10.1145/2591708.2591749.

Watson, C.; Li, F. W. B.; Godwin, J. L. (2013). Predicting performance in an introductory programming course by logging and analyzing student programming behavior. In *Proceedings - 2013 IEEE 13th International Conference on Advanced Learning Technologies, ICALT 2013* (pp. 319-323). doi:10.1109/ICALT.2013.99.

Yoo, J.; Kim, J. (2014). Can online discussion participation predict group project performance? Investigating the roles of linguistic features and participation patterns. *International Journal of Artificial Intelligence in Education*, 24(1), 8-32. doi:10.1007/s40593-013-0010-8.