

Predicción del Rendimiento Académico Universitario: Un Enfoque Multifactorial con Estadística Bayesiana. Primera Entrega "Implementación de la base de datos"

Pinzón, Ángela¹, Cervantes , Daniel²,
Yañez, Andrés³

Dpto. de Matemáticas
Universidad Externado de Colombia

Pregrado Ciencia de Datos

Curso de Bases de Datos
`angela.pinzon1@est.uexternado.edu.co`, `daniel.cervantes@est.uexternado.edu.co`,
`andres.yanez02@est.uexternado.edu.co`

September 11, 2023

Contents

1	Introducción	3
2	Características del proyecto de investigación que hace uso de Bases de Datos	3
2.1	Título del proyecto de investigación	3
2.2	Objetivo general	3
2.2.1	Objetivos específicos	3
2.3	Alcance	3
2.4	Pregunta de investigación	4
2.5	Hipótesis	4
3	Reflexiones sobre el origen de datos e información.	4
3.1	¿Cuál es el origen de los datos e información?	4
3.2	¿Cuáles son las consideraciones legales o éticas del uso de la información?	4
3.3	¿Cuáles son los retos de la información y los datos que utilizará en la base de datos en términos de la calidad y la consolidación?	5
3.4	¿Qué espera de la utilización de un sistema de Bases de Datos para su proyecto?	5
4	Diseño del Modelo de Datos del SMBD (Sistema Manejador de Bases de Datos)	5
4.1	Características del SMBD (Sistema Manejador de Bases de Datos) para el proyecto	5
4.2	Diagrama modelo de datos	5
4.3	Imágenes de la Base de Datos	5
4.4	Código SQL - lenguaje de definición de datos (DDL)	5
4.5	Código SQL - Manipulación de datos (DML)	8
4.6	Código SQL + Resultados: Vistas	10
4.7	Código SQL + Resultados: Triggers	10
4.8	Código SQL + Resultados: Funciones	11
4.9	Código SQL + Resultados: procedimientos almacenados	12

1 Introducción

El objetivo principal de este proyecto de investigación es determinar cómo una variedad de factores, como la duración del viaje desde el hogar hasta la universidad, el apoyo económico, el profesor a cargo, el conocimiento previo de la materia y otros, influyen en el rendimiento académico de los alumnos. A partir de esta determinación, se propone desarrollar un modelo estadístico con el fin de predecir el rendimiento académico de los estudiantes. Esta herramienta permitirá establecer rutas iniciales para alcanzar resultados académicos deseados, teniendo en cuenta las diferentes materias y profesores involucrados, entre otros factores.

Nuestro objetivo general es determinar cómo estos factores interactúan y afectan el rendimiento académico de los estudiantes universitarios y, a partir de esta comprensión, desarrollar un modelo estadístico efectivo que pueda predecir con precisión este rendimiento. La herramienta resultante no solo ofrecerá una visión integral de los determinantes académicos, si no que también nos dará datos importantes para la toma de decisiones frente a los aspectos académicos.

2 Características del proyecto de investigación que hace uso de Bases de Datos

2.1 Título del proyecto de investigación

El Título sugerido inicialmente por nosotros, sin tener definido inicialmente el modelo utilizado de estadística bayesiana: "Predicción del Rendimiento Académico Universitario: Un Enfoque Multifactorial con Estadística Bayesiana."

2.2 Objetivo general

Determinar cómo diversos factores, como la duración del viaje, el apoyo económico y otros, influyen en el rendimiento académico. Desarrollar un modelo estadístico para predecir el rendimiento de los estudiantes.

2.2.1 Objetivos específicos

- Identificar los factores determinantes del rendimiento académico: Se buscarán y determinarán los factores más influyentes en el rendimiento académico de los estudiantes.
- Analizar la relación entre factores y rendimiento académico: Se llevará a cabo un análisis estadístico de los datos recolectados para determinar la influencia de factores en el rendimiento académico de los alumnos.
- Proponer rutas de mejora académica: Utilizando el modelo desarrollado, se ofrecerán sugerencias concretas para mejorar el desempeño académico de los estudiantes.

2.3 Alcance

En un ambiente universitario, la búsqueda de métodos efectivos para mejorar el rendimiento académico de los estudiantes universitarios es una prioridad. Este proyecto de investigación surge con el propósito de abordar esta cuestión desde una perspectiva multifactorial y con el apoyo de

la estadística bayesiana. Nuestra investigación se centra en comprender cómo una amplia gama de factores, que van más allá de las aulas, influyen en el desempeño académico de los alumnos. Estos factores incluyen la duración del viaje desde el hogar hasta la universidad, el apoyo económico, el profesor a cargo de la materia, el conocimiento previo de la materia y otros elementos.

2.4 Pregunta de investigación

”¿Cómo pueden las variables como el apoyo económico, horas de estudio, profesor que dicta la materia, entre otras, influir en el rendimiento académico de los estudiantes de ciencia de datos de la universidad Externado de Colombia?”

2.5 Hipótesis

Al revisar otros estudios frente al tema, no encontramos modelos sencillos para lograr una predicción inicial ÚNICAMENTE con variables relacionadas directamente con las notas y las materias, es posible que el modelo crezca en complejidad proporsional a los datos que queramos utilizar y se deba investigar sobre técnicas o modelos muy específicos que se han usado para otros temas para realizar una extrapolación a nuestro problema.

3 Reflexiones sobre el origen de datos e información.

En esta sección, se abordarán cuestiones relacionadas con el origen de los datos e información que se utilizarán en la investigación, así como las consideraciones éticas y los desafíos asociados. También se discutirá la importancia de la calidad y consolidación de los datos y las expectativas relacionadas con el uso de un sistema de bases de datos como el paso inicial junto con la verificación del estado del arte.

3.1 ¿Cuál es el origen de los datos e información?

Los datos para esta investigación se obtendrán directamete de la informacion brindada por la universidad, incluyendo registros académicos de estudiantes, encuestas a alumnos, información sobre profesores y otros recursos institucionales. Mientras que por otro lado recopilamos otros estudios relacionados con la predición de notas académica, para la comprensión inicial del estado del arte frente a temas predictivos.

3.2 ¿Cuáles son las consideraciones legales o éticas del uso de la información?

El uso de datos académicos y personales implica consideraciones éticas y legales importantes. Se respetarán todas las regulaciones de privacidad y se obtendrá el consentimiento de los estudiantes cuando sea necesario para la recopilación de datos. Además, se protegerá la confidencialidad de la información personal y se seguirán los procedimientos éticos establecidos por la institución.

3.3 ¿Cuáles son los retos de la información y los datos que utilizará en la base de datos en términos de la calidad y la consolidación?

El mayor reto de la información es realizar de manera correcta las preguntas en el formulario que sera enviado a la institución. Hacer las preguntas correctas es muy importante para sacar información correcta de los datos.

3.4 ¿Qué espera de la utilización de un sistema de Bases de Datos para su proyecto?

El uso de un sistema de bases de datos permitirá una gestión eficiente de los datos recopilados, su almacenamiento seguro y la realización de consultas y análisis de manera efectiva. Se espera que el sistema de bases de datos facilite la manipulación de datos y contribuya a la robustez de la investigación al proporcionar una estructura organizada para el almacenamiento y acceso a la información. Inicialmente se plantea el uso de manejar un sistema estructurado de datos, sin embargo en la práctica creemos que sea mas versátil el uso de un sistema no estructurado, se determinara cual sistema servirá más en posteriores entregas.

4 Diseño del Modelo de Datos del SMBD (Sistema Manejador de Bases de Datos)

4.1 Características del SMBD (Sistema Manejador de Bases de Datos) para el proyecto

El modelo se creó originalmente en Oracle SQL Developer Data Modeler, sin embargo esta implementado en PostgreSQL. Que dentro de las características mas importantes que nos hizo elegir esta herramienta están:

- Fuerte soporte para consultas complejas y operaciones de base de datos.
- Extensiones propias con soporte de multiples lenguajes de programación, incluidos python, lo que nos permite interactuar con código en python y usar lenguaje SQL dentro de este.

4.2 Diagrama modelo de datos

4.3 Imágenes de la Base de Datos

4.4 Código SQL - lenguaje de definición de datos (DDL)

```
1 CREATE TABLE alumno (  
2     id_alumno    INTEGER NOT NULL ,  
3     promedio     FLOAT ,  
4     semestre     INTEGER ,  
5     anio_entrada INTEGER ,  
6     sexo         VARCHAR(10) ,  
7     edad         INTEGER
```

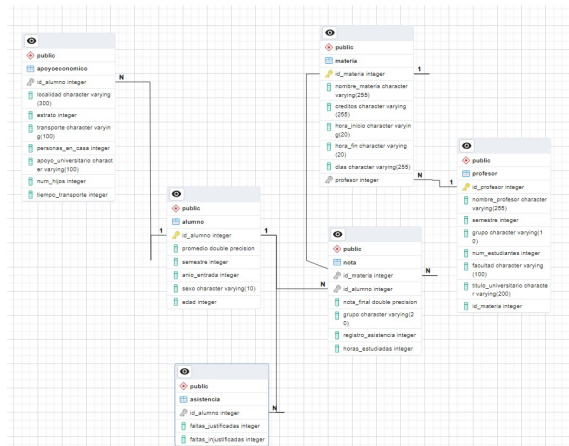


Figure 1: Diagrama modelo de datos

	id_alumno [PK] integer	promedio double precision	semestre integer	anio_entrada integer	sexo character varying (10)	edad integer								
1	1	2.713090877	8	2021	Femenino	22	id_alumno integer	localidad character varying (300)	estrato integer	transporte character varying (100)	personas_en_casa integer	apoyo_universitario character varying (100)	num_hijos integer	tiempo_transporte integer
2	2	0.19	6	2006	Masculino	21	1	Josechester	5	Ruta	6	Si	4	15
3	3	2.86935941356	7	2001	Femenino	29	2	Port Charles	4	Caminando	2	No	2	66
4	4	0.982	6	2003	Masculino	23	3	Rivasmouth	6	Taxi	2	No	0	96
5	5	2	2	2008	Masculino	23	4	Daltonberg	2	Servicios de transporte privados	6	No	5	25
6	6	4.6585679140352	4	2016	Masculino	29	5	Sharontown	6	Bicicleta	2	No	2	19
7	7	4.719	2	2019	Femenino	21	6	Reginaside	5	Taxi	1	Si	5	87
8	8	1.629463	2	2007	Femenino	19	7	Mariland	1	Bus	9	No	5	113
							8	Salazartown	2	Taxi	5	No	3	111

Figure 2: BD Alumnos

Figure 3: BD Apoyo Económico

```

8 );
9
10 ALTER TABLE alumno
11 ADD CONSTRAINT alumno_pk
12 PRIMARY KEY ( id_alumno );
13
14 CREATE TABLE apoyoeconomico (
15     id_alumno                INTEGER NOT NULL,
16     localidad                 VARCHAR(300),
17     estrato                   INTEGER,
18     transporte                 VARCHAR(100),
19     personas_en_casa          INTEGER,
20     apoyo_universitario       FLOAT,
21     num_hijos                  INTEGER,
22     tiempo_transporte         INTEGER
23 );
24
25 CREATE TABLE asistencia (

```

id_materia [PK] integer	nombre_materia character varying (255)	creditos integer	hora_inicio character varying (20)	hora_fin character varying (20)	dias character varying (255)	profesor integer	id_materia integer	id_alumno integer	nota_final double precision	grupo character varying (20)	registro_asistencia integer	horas_estudiadas integer
1	Adaptive user-facing Graphic Interface	3	03:54AM	03:54AM	Martes-Jueves	1	1	1	4.457688417520725	C		4
2	Pre-emptive systematic Graphic Interface	2	10:43AM	10:43AM	Viernes	2		2	0.6715238546009383	A		8
3	Multi-layered bifurcated middleware	3	02:37AM	02:37AM	Viernes	3	2	2			6	
4	User-centric contextually-based project	4	07:20AM	07:20AM	Martes-Jueves	4	3	3	2.3505290369655167	C		10
5	De-engineered encompassing frame	4	02:58AM	02:58AM	Martes-Jueves	5		4	2.802800707595861	B		10
6	Proactive systemic infrastructure	1	02:30AM	02:30AM	Martes-Jueves	6	5	5	1.46692651445731	A		7
7	Reverse-engineered responsive monitoring	3	09:58AM	09:58AM	Viernes	7		6	2.2139342270678846	C		9
8	Open-source static projection	2	03:22AM	03:22AM	Viernes	8	7	7	4.979553178702773	B		5
9	Integrated multimedia neural-net	1	10:20AM	10:20AM	Martes-Jueves	9					8	
10	Advanced bi-directional process improvement	1	09:00AM	09:00AM	Lunes-Miércoles	10						

Figure 4: BD Materia

Figure 5: BD Notas

id_profesor [PK] integer	nombre_profesor character varying (255)	semestre integer	grupo character varying (10)	num_estudiantes integer	facultad character varying (100)	titulo_universitario character varying (200)	id_materia integer
1	Shawn Robinson	7	B		11	Materia G	1
2	Kristen Henderson	3	C		18	Materia D	2
3	Joyce Stewart	1	B		25	Materia B	3
4	Richard Dunn	9	A		11	Materia D	4
5	Terri Nielsen	10	B		37	Materia C	5
6	Rebecca Diaz	5	C		34	Materia D	6
7	Mark Nguyen	7	C		21	Materia D	7
8	Tiffany Williams	4	B		19	Materia H	8
9	Bethany Parks	8	A		33	Materia G	9
10	Brandon Robertson	2	C		25	Materia D	10

Figure 6: BD Profesor

```

26     id_alumno          INTEGER NOT NULL,
27     faltas_justificadas INTEGER,
28     faltas_injustificadas INTEGER
29 );
30
31 CREATE TABLE materia (
32     id_materia          INTEGER NOT NULL,
33     nombre_materia      VARCHAR(255),
34     creditos            VARCHAR(255),
35     horario             TIMESTAMP,
36     profesor            INTEGER NOT NULL
37 );
38
39 ALTER TABLE materia
40 ADD CONSTRAINT materia_pk
41 PRIMARY KEY ( id_materia );
42
43 CREATE TABLE nota (
44     id_materia          INTEGER NOT NULL,
45     id_alumno           INTEGER NOT NULL,
46     nota_final          FLOAT,
47     grupo               VARCHAR(20),
48     horario             TIMESTAMP,
49     registro_asistencia INTEGER,
50     horas_estudiadas    INTEGER
51 );

```

```

52
53 CREATE TABLE profesor (
54     id_profesor          INTEGER NOT NULL,
55     nombre_profesor      VARCHAR(255),
56     semestre             INTEGER,
57     grupo                VARCHAR(10),
58     num_estudiantes      INTEGER,
59     facultad             VARCHAR(100),
60     titulo_universitario VARCHAR(200),
61     id_materia           INTEGER
62 );
63
64 ALTER TABLE profesor
65 ADD CONSTRAINT profesor_pk
66 PRIMARY KEY ( id_profesor );
67
68 ALTER TABLE asistencia
69     ADD CONSTRAINT asistencia_alumno_fk
70     FOREIGN KEY ( id_alumno )
71     REFERENCES alumno ( id_alumno );
72
73 ALTER TABLE materia
74     ADD CONSTRAINT table_3_profesor_fk
75     FOREIGN KEY ( profesor )
76     REFERENCES profesor ( id_profesor );
77
78 ALTER TABLE apoyoeconomico
79     ADD CONSTRAINT table_6_alumno_fk
80     FOREIGN KEY ( id_alumno )
81     REFERENCES alumno ( id_alumno );
82
83 ALTER TABLE nota
84     ADD CONSTRAINT table_7_alumno_fk
85     FOREIGN KEY ( id_alumno )
86     REFERENCES alumno ( id_alumno );
87
88 ALTER TABLE nota
89     ADD CONSTRAINT table_7_materia_fk
90     FOREIGN KEY ( id_materia )
91     REFERENCES materia ( id_materia );

```

4.5 Código SQL - Manipulación de datos (DML)

```

1 import psycopg2
2 from faker import Faker

```



```

3
4 dbname = "NotesStudy"
5 user = "postgres"
6 password = "1234"
7 host = "localhost"
8 port = "5432"
9
10 faker = Faker()
11
12 # Generar 100 datos de prueba para cada tabla
13 alumnos = [(i, faker.pyfloat(min_value=0, max_value=5), faker.random_int(min=1, max=8),
14             faker.random_int(min=2000, max=2022), faker.random_element(["Masculino", "Femenino"]),
15             faker.random_int(min=16, max=26)) for i in range(1, 101)]
16
17 apoyos = [(i, faker.city(), faker.random_int(min=1, max=6), faker.random_element(["Bus", "Metro", "Bicicleta"]),
18            faker.random_int(min=1, max=6), faker.random_int(min=0, max=10000000),
19            faker.random_int(min=0, max=2), faker.random_int(min=5, max=120)) for i in range(1, 101)]
20
21 asistencias = [(i, faker.random_int(min=0, max=20), faker.random_int(min=0, max=20)) for i in range(1, 101)]
22
23 materias = [(i, faker.catch_phrase(), faker.random_int(min=1, max=5),
24              faker.time_series(start_date='-30d', end_date='now', precision=None, distrib=None), faker.random_int(
25
26 profesores = [(i, faker.name(), faker.random_int(min=1, max=8), faker.random_element(["A", "B", "C"]),
27                faker.random_int(min=1, max=50), faker.job(), faker.job(),
28                faker.random_int(min=1, max=100)) for i in range(1, 101)]
29
30 notas = [(i, j, faker.pyfloat(min_value=0, max_value=5),
31            faker.random_element(["A", "B", "C"]), faker.date_time_this_year(),
32            faker.random_int(min=1, max=10), faker.random_int(min=1, max=10)) for i in range(1, 101) for j in range(1, 101)]
33
34 # Insertar datos de prueba en la base de datos
35 try:
36     connection = psycopg2.connect(
37         dbname=dbname, user=user, password=password, host=host, port=port
38     )
39
40     cursor = connection.cursor()
41
42     # Insertar datos en la tabla alumno
43     insert_query = "INSERT INTO alumno (id_alumno, promedio, semestre, anio_entrada, sexo, edad) VALUES (%s, %s, %s, %s, %s, %s)"
44     cursor.executemany(insert_query, alumnos)
45     # Insertar datos en la tabla apoyoeconomico
46     insert_query = "INSERT INTO apoyoeconomico (id_alumno, localidad, estrato, transporte, personas_en_casa, ap
47     cursor.executemany(insert_query, apoyos)
48     # Insertar datos en la tabla asistencia
49     insert_query = "INSERT INTO asistencia (id_alumno, faltas_justificadas, faltas_injustificadas) VALUES (%s, %s, %s)"
50     cursor.executemany(insert_query, asistencias)
51     # Insertar datos en la tabla profesor
52     insert_query = "INSERT INTO profesor (id_profesor, nombre_profesor, semestre, grupo, num_estudiantes, facul

```

```

53     cursor.executemany(insert_query, profesores)
54     # Insertar datos en la tabla materia
55     insert_query = "INSERT INTO materia (id_materia, nombre_materia, creditos, horario, profesor) VALUES (%s, %s, %s, %s, %s)"
56     cursor.executemany(insert_query, materias)
57     # Insertar datos en la tabla nota
58     insert_query = "INSERT INTO nota (id_materia, id_alumno, nota_final, grupo, horario, registro_asistencia, h) VALUES (%s, %s, %s, %s, %s, %s, %s)"
59     cursor.executemany(insert_query, notas)
60     connection.commit()
61     cursor.close()
62
63 except psycopg2.Error as e:
64     print("Error al conectarse a la base de datos:", e)

```

Listing 1: Código SQL - Manipulación de Datos (DML) para la tabla 'nota'

4.6 Código SQL + Resultados: Vistas

```

1 CREATE VIEW promedio_notas_por_materia AS
2 SELECT id_materia, AVG(nota_final) as promedio_nota
3 FROM nota
4 GROUP BY id_materia;
5
6 CREATE VIEW alumnos_con_apoyo_y_promedio AS
7 SELECT a.id_alumno, a.promedio, ae.localidad, ae.apoyo_universitario, ae.num_hijos,
8 FROM alumno a
9 JOIN apoyoeconomico ae ON a.id_alumno = ae.id_alumno;
10
11 CREATE VIEW materias_y_profesores AS
12 SELECT m.nombre_materia, p.nombre_profesor
13 FROM materia m
14 JOIN profesor p ON m.profesor = p.id_profesor;

```

4.7 Código SQL + Resultados: Triggers

```

1 CREATE OR REPLACE TRIGGER actualizar_promedio
2 AFTER INSERT ON nota
3 FOR EACH ROW
4 BEGIN
5     UPDATE alumno
6     SET promedio = (SELECT AVG(nota_final)
7     FROM nota
8     WHERE id_alumno = :NEW.id_alumno)
9     WHERE id_alumno = :NEW.id_alumno;
10 END;
11

```

```

12 CREATE OR REPLACE TRIGGER verificar_edad
13 BEFORE INSERT ON alumno
14 FOR EACH ROW
15 BEGIN
16     IF :NEW.edad < 18 THEN
17         RAISE_APPLICATION_ERROR(-20001, 'El alumno debe ser mayor de 15 anos.');
```

4.8 Código SQL + Resultados: Funciones

```

1 CREATE OR REPLACE FUNCTION calcular_edad(fecha_nacimiento DATE) RETURN INTEGER AS
2     edad INTEGER;
3 BEGIN
4     SELECT FLOOR(MONTHS_BETWEEN(SYSDATE, fecha_nacimiento) / 12) INTO edad FROM DUAL;
5     RETURN edad;
6 END;
7
8 CREATE OR REPLACE FUNCTION obtener_promedio_alumno(p_id_alumno INTEGER) RETURN FLOAT
9     promedio FLOAT;
10 BEGIN
11     SELECT AVG(nota_final) INTO promedio
12     FROM nota
13     WHERE id_alumno = p_id_alumno;
14     RETURN promedio;
15 END;
16
17 CREATE OR REPLACE FUNCTION tiene_faltas_injustificadas(p_id_alumno INTEGER) RETURN
18     faltas INTEGER;
```

```

19 BEGIN
20     SELECT faltas_injustificadas INTO faltas
21     FROM asistencia
22     WHERE id_alumno = p_id_alumno;
23
24     RETURN faltas > 0;
25 END;
26
27 CREATE OR REPLACE FUNCTION obtener_nombre_profesor(p_id_materia INTEGER) RETURN VARCHAR(255)
28     nombre_profesor VARCHAR(255);
29 BEGIN
30     SELECT p.nombre_profesor INTO nombre_profesor
31     FROM profesor p
32     JOIN materia m ON p.id_profesor = m.profesor
33     WHERE m.id_materia = p_id_materia;
34
35     RETURN nombre_profesor;
36 END;

```

4.9 Código SQL + Resultados: procedimientos almacenados

```

1 CREATE OR REPLACE PROCEDURE calcular_promedio_alumno (p_id_alumno IN INTEGER, p_promedio OUT REAL)
2 BEGIN
3     SELECT AVG(nota_final) INTO p_promedio
4     FROM nota
5     WHERE id_alumno = p_id_alumno;
6 END;
7
8 CREATE OR REPLACE PROCEDURE actualizar_promedio_alumno (p_id_alumno IN INTEGER) AS
9 BEGIN
10     UPDATE alumno
11     SET promedio = (SELECT AVG(nota_final)
12     FROM nota WHERE id_alumno = p_id_alumno)
13     WHERE id_alumno = p_id_alumno;
14 END;
15
16 CREATE OR REPLACE PROCEDURE insertar_nota (p_id_materia IN INTEGER, p_id_alumno IN INTEGER, p_nota_final IN REAL)
17 BEGIN
18     INSERT INTO nota (id_materia, id_alumno, nota_final, grupo, horario, registro_a, registro_b, registro_c)
19     VALUES (p_id_materia, p_id_alumno, p_nota_final, NULL, SYSTIMESTAMP, NULL, NULL, NULL);
20     COMMIT;
21 END;
22
23 CREATE OR REPLACE PROCEDURE obtener_nombre_profesor (p_id_materia IN INTEGER, p_nombre OUT VARCHAR(255))
24 BEGIN

```

```
25 SELECT p.nombre_profesor INTO p_nombre_profesor
26 FROM profesor p
27 JOIN materia m ON p.id_profesor = m.profesor
28 WHERE m.id_materia = p_id_materia;
29 END;
```