



**Corporación Unificada Nacional  
de Educación Superior**

VIGILADA MINEDUCACIÓN

Optimización del perfil estudiantil para obtener un puesto laboral

Investigación de operaciones

Andrés Felipe Carmona Orozco

Juan Jose Rincon Gómez

Leidy Yulieth Méndez Galeano

Septiembre 2025

## Contenido

<b>Introducción .....</b>	<b>3</b>
<b>Marco Teórico .....</b>	<b>4</b>
<b>Objetivos .....</b>	<b>5</b>
<b>Objetivo general .....</b>	<b>5</b>
<b>Objetivos específicos .....</b>	<b>5</b>
<b>Metodología .....</b>	<b>6</b>
<b>Datos .....</b>	<b>7</b>
<b>Análisis de datos .....</b>	<b>8</b>
<b>Histograma de CGPA .....</b>	<b>9</b>
<b>Grafico de barras de experiencia en practicas .....</b>	<b>10</b>
<b>Gráfico de barras de cantidad de estudiantes con un puesto laboral.....</b>	<b>11</b>
<b>Gráfico de dispersión CGPA contra Habilidades Comunicativas coloreando si han obtenido un puesto laboral .....</b>	<b>12</b>
<b>Modelo de optimización.....</b>	<b>13</b>
<b>Función .....</b>	<b>13</b>
<b>Restricciones .....</b>	<b>14</b>
<b>Resultados .....</b>	<b>15</b>
<b>Referencias.....</b>	<b>17</b>
<b>Anexos .....</b>	<b>18</b>

## **Introducción**

Una de las principales razones por la que las personas continúan su proceso de aprendizaje luego de finalizar su educación además de ser un reto personal, en algunos casos también lo hacen con el fin de encontrar un puesto laboral; Encontrar un puesto laboral para los estudiantes universitarios depende de distintos factores tanto académicos como personales, estos abarcan el rendimiento académico, las habilidades de comunicación, si ya cuenta con una experiencia que van a reforzar, etc.

Por medio de este informe se busca analizar estos factores utilizando herramientas de investigación de operaciones, como la programación lineal, para identificar la mejor combinación de atributos que aumente la empleabilidad de los estudiantes.

## Marco Teórico

### - Empleabilidad universitaria

La empleabilidad se refiere a la capacidad de una persona para obtener y mantener un empleo, así como para adaptarse a los cambios del mercado laboral. En el contexto universitario, esta se ve influenciada por factores académicos como el rendimiento académico (GPA, CGPA), la participación en proyectos, las habilidades comunicativas y la experiencia en prácticas profesionales. Según Rodríguez y Pérez (2020), fortalecer estos aspectos desde la formación académica mejora significativamente las oportunidades de inserción laboral de los egresados.

### - Investigación de Operaciones (IO)

La Investigación de Operaciones es una disciplina que utiliza modelos matemáticos y técnicas analíticas para apoyar la toma de decisiones en sistemas complejos. En el ámbito educativo, permite analizar y optimizar procesos relacionados con la formación, la asignación de recursos y la evaluación de perfiles estudiantiles. Su aplicación en problemas reales facilita la identificación de soluciones eficientes y prácticas (UNADMéxico, 2020).

### - Programación Lineal

La programación lineal es una herramienta matemática que permite encontrar la mejor solución posible a un problema de optimización, sujeto a restricciones lineales. Su objetivo es maximizar o minimizar una función objetivo, como puede ser la probabilidad de colocación laboral. El método simplex, ampliamente utilizado en esta técnica, permite resolver modelos con múltiples variables y restricciones. En contextos educativos y empresariales, la programación lineal ha demostrado ser útil para distribuir recursos, mejorar procesos y tomar decisiones estratégicas (Universidad San Ignacio de Loyola, 2019).

## **Objetivos**

### **Objetivo general**

El objetivo general del informe es analizar los factores que influyen a que los estudiantes universitarios puedan conseguir un puesto laboral de manera mas efectiva, utilizando herramientas de investigación de operaciones y proponer mejoras de empleabilidad en estos.

### **Objetivos específicos**

- Examinar los datos para identificar que variables son mas relevantes y pueden influir en la obtención de un puesto laboral.
- Aplicar un modelo de programación lineal que nos permita encontrar la mejor combinación de factores que influyan en la obtención de un puesto laboral

## Metodología

Todo este trabajo se desarrollará de forma cuantitativa, buscando analizar y optimizar los factores que influyen en la obtención de un puesto laboral por parte de los estudiantes universitarios. Para realizar dichas tareas se utilizarán herramientas básicas de investigación de operaciones, específicamente la programación lineal, con el fin de identificar la combinación de atributos que ayuden a aumentar obtención de un puesto laboral.

Para desarrollar dicho modelo se realizara por medio del lenguaje de programación en Python, se va a desarrollar en este lenguaje debido a la cantidad de datos que deben ser analizados y porque tiene una gran flexibilidad en cuanto a la solución de modelos, adicionalmente se utilizara la biblioteca PuLP, ya que esta nos va a facilitar la formulación del problema, ya que dicha biblioteca o librería nos permite definir variables de decisión, establecer funciones objetivo, establecer restricciones y resolver el modelo utilizando el método simplex además utilizaremos la librería Scikit-learn que es una librería utilizada en el Machine Learning que nos permitirá predecir la mejor combinación aprendiendo de las habilidades de las personas que si cuentan con un puesto laboral también se utilizara la librería de pandas para poder manipular nuestros datos y por ultimo utilizaremos ya que Scikit-learn necesita de ella porque nos permite dar estructuras a los datos.

El modelo se basa a partir de variables como el GPA acumulado, la experiencia en prácticas profesionales, las habilidades comunicativas, el número de proyectos completado y la participación en actividades extracurriculares; Estas variables se representan como  $x_1, x_2, x_3, x_4, x_5$ , con el fin de entender mejor el modelo a usar. La función objetivo busca maximizar la probabilidad de la obtención de un puesto laboral y se expresa como una combinación lineal ponderada, que representa la importancia de cada factor y seria

$$\text{Maximizar } Z = c_1x_1 + c_2x_2 + c_3x_3 + c_4x_4 + c_5x_5$$

Una vez ya realizada la formulación del modelo se procede a la resolución mediante el método simplex, que nos permitirá encontrar la solución óptima, ya que este método es muy utilizado por su precisión y rapidez.

## Datos

El conjunto de datos que va a ser utilizado para este informe se extrajo de la plataforma Kaggle, específicamente del repositorio con título College Student Placement Factors Dataset. Que cuenta con 10.000 datos simulados de perfiles académicos y profesionales de estudiantes universitarios.

La estructura de los datos seleccionados varia ya que contiene variables tanto numéricas como de texto, pero para efectos de análisis se usarán numéricas remplazando las de texto por “1” en caso de que sea “Si” o “0” en caso de que sea “No”.

Las variables que incluye el conjunto de datos son:

- College\_ID: Identificación única de la universidad (por ejemplo, CLG0001 a CLG0100)
- IQ: Puntuación de CI del estudiante (normalmente distribuida alrededor de 100).
- Prev\_Sem\_Result: Promedio de calificaciones (GPA) del semestre anterior (rango: 5.0 a 10.0).
- CGPA: Promedio de calificaciones acumulativo (rango: ~5.0 a 10.0).
- Academic\_Performance: Calificación académica anual (escala: 1 a 10).
- Internship\_Experience: Si el estudiante ha realizado alguna pasantía o practicas (Sí/No) para este caso como se menciono anteriormente esta variable en el conjunto de datos será remplazada por “0” y “1” según corresponda.
- Extra\_Curricular\_Score: Participación en actividades extracurriculares (puntuación de 0 a 10)
- Communication\_Skills: Calificación de habilidades blandas (escala: 1 a 10)
- Projects\_Completed: Número de proyectos académicos/técnicos completados (0 a 5)
- Placement: Resultado final de la colocación (Sí = Colocado, No = No Colocado) para este otro caso como se mencionó anteriormente esta variable en el conjunto de datos será remplazada por “0” y “1” según corresponda.

Gracias a este conjunto de datos podemos construir un modelo que nos permita maximizar la probabilidad de que un universitario obtenga un puesto laboral, la cantidad y diversidad de datos hacen que sean perfectos para aplicar una optimización y análisis a dichos datos.

## **Análisis de datos**

Antes de realizar el modelo vamos a realizar el Análisis de datos, esto lo realizamos debido a que nos permite identificar patrones o tendencias que mas pueden influir en la obtención de un puesto laboral. Para realizar este análisis de datos se van a utilizar graficas simples y muy claras que facilitan interpretar la información.

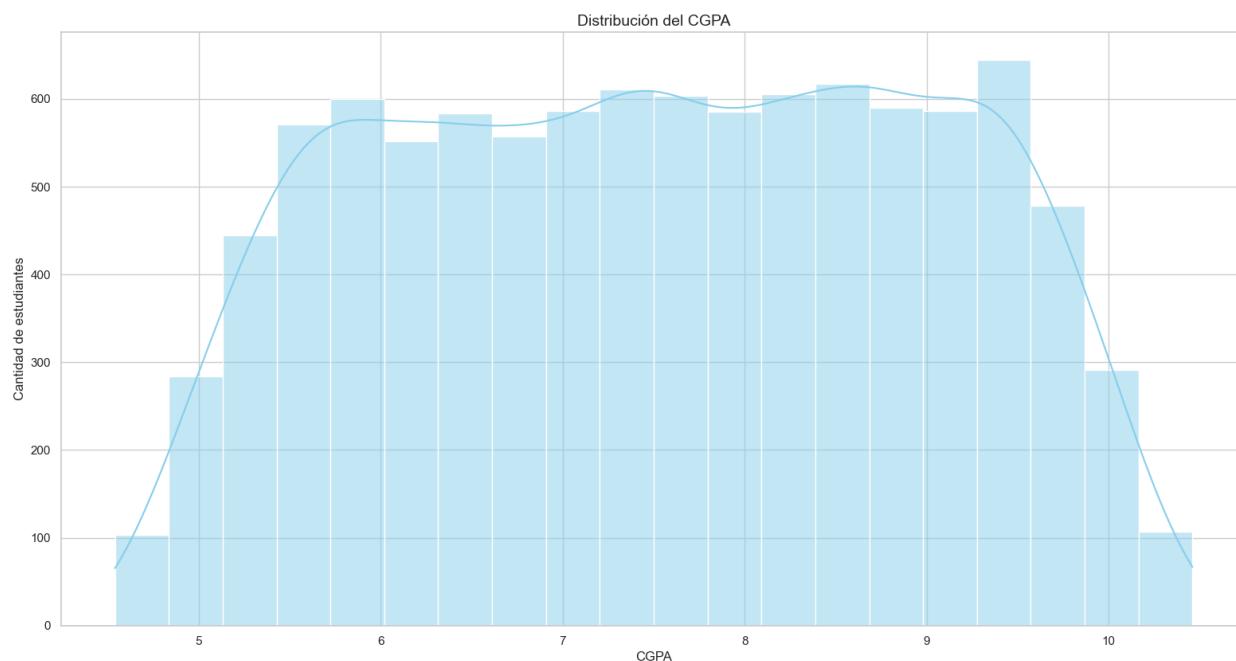
Todas estas graficas se van a realizar en Python utilizando librerías como Pandas que es muy reconocido y muy usado para este tipo de casos ya que nos permite cargar nuestra información, limpiarla y manipularla, también utilizaremos Matplotlib ya que esta nos da una base de los gráficos y nos permite personalizar nuestros gráficos y por último utilizaremos Seaborn que es una librería que trabaja de la mano Matplotlib ya que nos permite darle estilos y además también de la mano de Pandas ya que necesita las estructuras de los datos que nos permite manipular los datos.

Las gráficas van a mostrar la distribución, dispersión y como se pueden relacionar unas variables con otras para la obtención de un puesto laboral.



## Histograma de CGPA

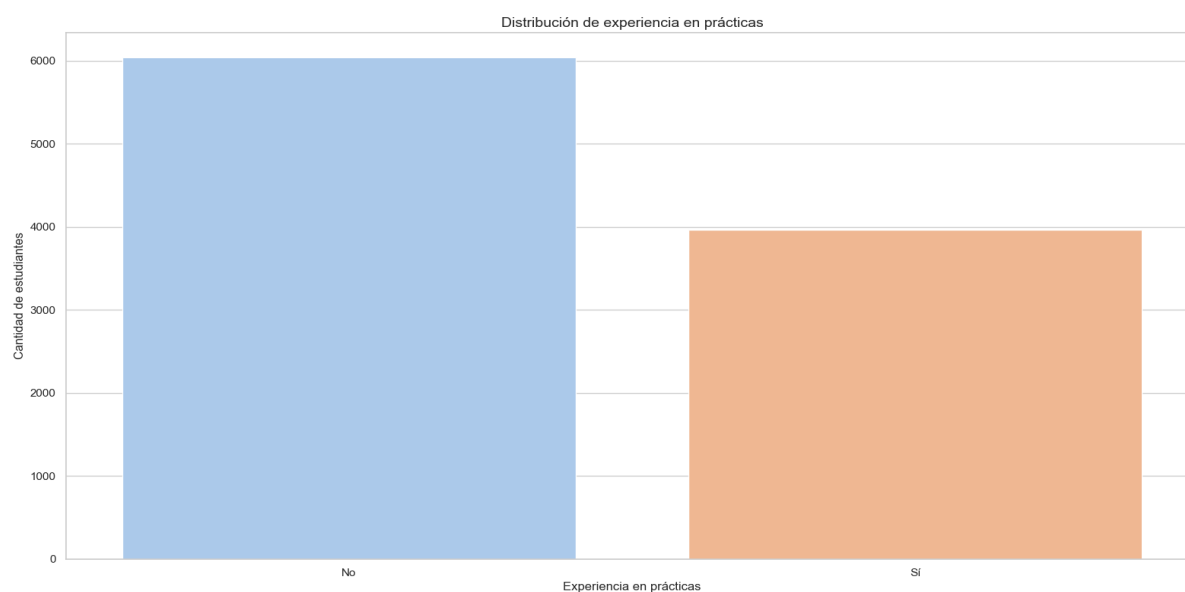
Se decidió realizar un histograma del CGPA o promedio de calificaciones acumulativo debido a que este es un indicador clave del rendimiento académico de los estudiantes y por medio de este se puede identificar si la mayoría de estudiantes están entre un rango bajo, medio o alto.



Gracias a esta grafica podemos deducir que la gran mayoría de notas están distribuidas en un rango entre 5 y 9 lo que indica que la mayoría de estudiantes en base a sus notas tienen un rendimiento entre medio y alto. Lo que nos puede ayudar a deducir si el rendimiento académico podría influir en la obtención de un puesto laboral.

## Grafico de barras de experiencia en practicas

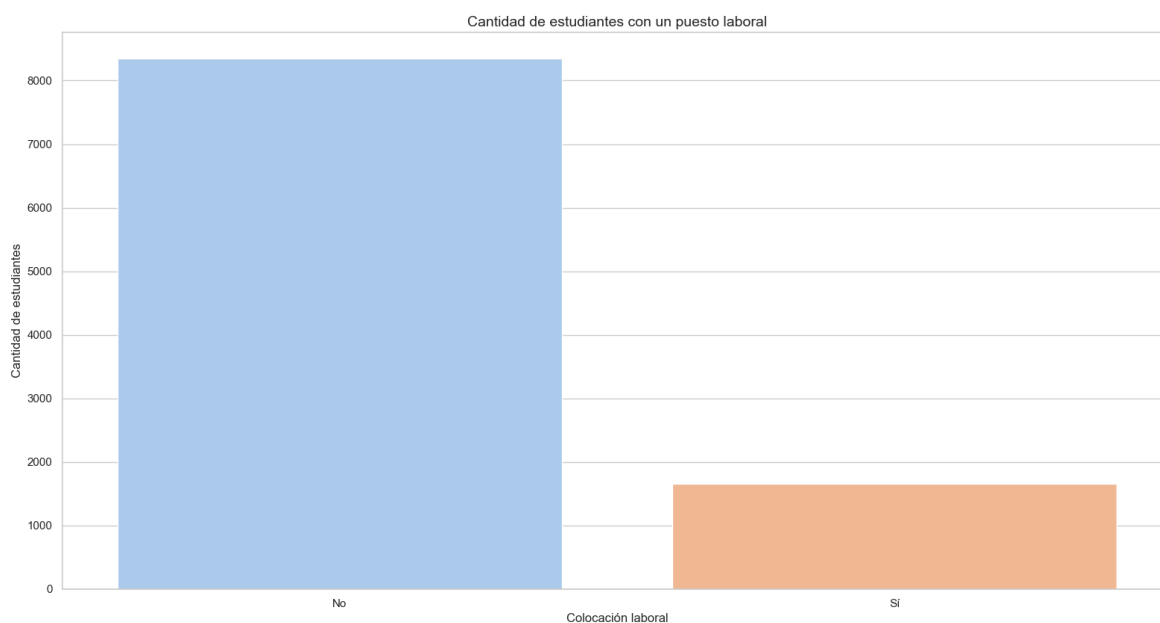
Esta grafico se realiza con la intención de saber la cantidad de estudiantes que tienen o no tienen una experiencia laboral previa lo cual puede ser un gran determinante para este caso y por medio de este grafico se puede ver la cantidad de estudiantes que cuentan y no cuentan con ella.



El resultado de esta grafica nos arroja que la gran mayoría de estudiantes no cuentan con una experiencia laboral previa lo cual nos va a permitir evaluar posteriormente si esto influye o no influye en la obtención de un puesto laboral.

### Gráfico de barras de cantidad de estudiantes con un puesto laboral

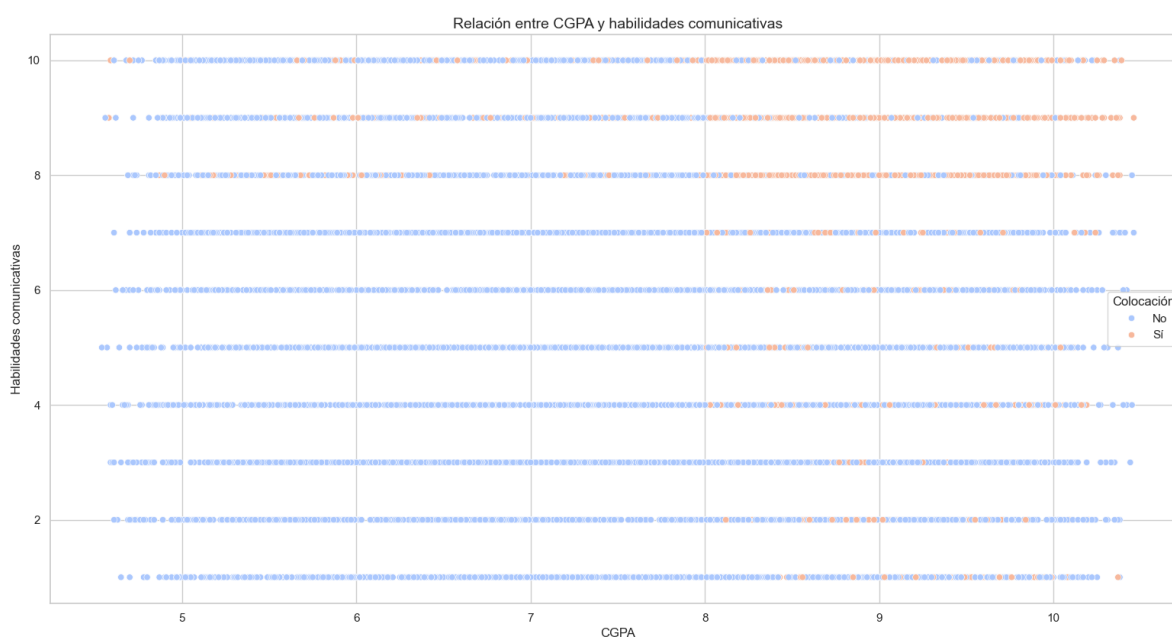
Este grafico se seleccionó para ver la cantidad de estudiantes que si cuentan con un puesto laboral y ver si existe un desequilibrio entre la cantidad de estudiantes que si tienen un puesto laboral y los que no lo han conseguido y así poder realizar un mejor análisis.



Como se puede observar en la gráfica la gran mayoría de estudiantes no han obtenido un puesto laboral mientras que por otro lado un porcentaje más pequeño si cuentan con uno, y gracias a esto podemos verificar que estudiantes si lo han obtenido y poder ver que influye en la obtención de este.

## Gráfico de dispersión CGPA contra Habilidades Comunicativas coloreando si han obtenido un puesto laboral

Para finalizar este grafico se busca mirar si hay una relación entre el rendimiento académico y las habilidades comunicativas que ha menudo y en gran parte de empresas se evalúan para saber si se contrata o no a una persona, ya que es de gran importancia en este caso saber si no solo se es un buen estudiante, sino que también es capaz de comunicarse de una forma clara y precisa.



Para entender este grafico no solo es necesario poner atención a los dos ejes de CGPA y Habilidades comunicativas sino también al color de cada circulo.

Como se puede observar en el grafico a pesar de que hay casos en la que un CGPA alto y pocas habilidades comunicativas no influyen en la obtención de un puesto laboral, la gran mayoría de personas que cuentan con este tienen un CGPA alto y un alto nivel en habilidades comunicativas lo que nos puede ayudar a deducir que tal vez estos factores tengan una gran importancia para la obtención de un puesto laboral.

## Modelo de optimización

Luego de realizar el análisis de los datos pasamos a hacer el modelo de optimización basado en programación lineal para poder identificar que combinación maximiza la probabilidad de que un estudiante obtenga un puesto laboral. Por medio de este modelo buscamos recomendar las actividades a las cuales los estudiantes deben darle mas importancia para obtener un puesto laboral.

Para realizar el modelo usaremos lo mencionado en la metodología que es Python, PuLP, Scikit-learn, Pandas y NumPy que son librerías que nos ayudaran a manipular y obtener los resultados luego de realizar el análisis por medio de funciones con las que ellas cuentan.

## Función

La función que se busca es maximizar la optencion de un puesto laboral, que esto se obtiene por medio de la combinación de los factores que pueden tener importancia en este caso, la expresaríamos como.

$$\text{Maximizar } Z = c_1x_1 + c_2x_2 + c_3x_3 + c_4x_4 + c_5x_5$$

Donde las  $x$  representan los valores de los factores es decir el valor asignado a CGPA, Internship\_Experience, Extra\_Curricular\_Score, Communication\_Skills y Projects\_Completed respectivamente mientras que las  $c$  representan qué importancia tiene esa habilidad para obtener un puesto laboral.

## Restricciones

Las restricciones que usaremos en este caso serán las siguientes:

**CGPA mayor o igual 8.0:** Esta restricción se va a usar ya que una gran parte de los estudiantes alcanza este rango por lo tanto se podrá abarcar mejor el estudio; Es decir  $x_1 = 1$  si  $CGPA \geq 8.0$ .

**Habilidades comunicativas mayor o igual a 7.0:** Esta restricción se basa en que la mayoría de estudiantes a menudo no son buenos comunicadores y según la gráfica de dispersión una gran parte se encuentran en este rango por lo que relativamente es posible alcanzar; Es decir  $x_3 = 1$  si  $Communication\_Skills \geq 7.0$ .

**Carga académica menor o igual a 3:** Esta restricción es para darle mas realidad a nuestro modelo debido a que un estudiante no va a tener el suficiente tiempo para mejorar todas las habilidades por eso en este caso solo se podrán realizar mejoras en 3 habilidades que por medio de los datos el modelo decidirá cuales son más importantes e influyen; Es decir  $x_1 + x_2 + x_3 + x_4 + x_5 \leq 3$ .

**Practicas respecto a CGPA:** Por medio de esta restricción buscamos también darle mas realidad al modelo, ya que una persona con un promedio bajo no debería realizar practicas profesionales, ya que cuenta con conocimientos bajos para realizar o ayudar en una práctica profesional; Es decir  $x_2 < x_1$

**Restricción mínima respecto a CGPA o Puntuación de Proyectos:** Por medio de esta restricción buscamos que se pueda maximizar la obtención, ya que en el análisis de datos anterior miramos que el CGPA alto tiene relación con la obtención de un puesto laboral y para compensar la disminución del CGPA utilizamos la puntuación en proyectos académicos que pueden tener mucho valor y aprendizaje para obtener al menos una aptitud que puede ser de gran importancia; Es decir  $x_1 + x_4 = 1$ .

Todas estas restricciones serán añadidas a la hora de crear el código en Python y se podrán visualizar.

## Resultados

Luego de realizar el modelo en Python y múltiples cálculos ([Ver Anexo D](#)), el resultado obtenido respecto a los datos es el siguiente

```
Estado de la solucion: Optimal

Habilidades para maximizar la obtencion de un puesto laboral:
Actividades_Extracurriculares_Altas = 0.0
CGPA_Alto = 1.0
Experiencia_Practicas = 0.0
Habilidades_Comunicativas_Altas = 1.0
Proyectos_Completados_Altos = 1.0

Z =: 0.9872
Porcentaje de obtencion de un puesto laboral: 98.72%

Recomendacion para obtener un puesto laboral
- Debe mantener un CGPA >= 8.0.
- Debe desarrollar habilidades comunicativas >= 7.0.
- Debes completar >= 3 proyectos academicos.
```

Se puede observar que el modelo encontró una combinación óptima que maximiza la obtención de un puesto laboral.

Lo primero que se puede observar es que nos da una lista de las habilidades que el modelo considera necesaria para la obtención de un puesto laboral en 0 y 1 recordemos que 0 equivale a no y 1 a si por lo tanto el modelo nos muestra que un CGPA alto o en el límite, habilidades comunicativas altas o en el límite y realizar proyectos académicos son de gran importancia para la obtención de un puesto laboral cabe resaltar que las experiencias en prácticas y las actividades extracurriculares no sean importes, sino que debido a la restricción añadida para el modelo tiene mucho más valor estas 3 habilidades.

Por otro lado, nos muestra el valor de Z, que multiplicado por 100 nos quiere decir el porcentaje de combinar estas habilidades para obtener un puesto laboral en este caso, al realizar esa combinación se obtiene una probabilidad del 98.72% de obtener un puesto laboral.

Por último, nos muestra de una manera más visual y en base al primer resultado y a las restricciones cuales son esas habilidades que cuentan con mas peso respecto a los datos suministrados y a las restricciones, que son mantener un CGPA mayor o igual a 8, tener una calificación mayor o igual a 7 en habilidades comunicativas y realizar 3 o más proyectos académicos.

## Conclusiones

- Gracias al análisis realizado y al modelo que se realizó, se logró identificar las habilidades o factores que mas influyen a la hora de obtener un puesto laboral por parte de los estudiantes. Por medio de herramientas de investigación de operaciones siendo más específico la programación lineal, se identifico que las habilidades o factores como CGPA, habilidades comunicativas y la realización de proyectos académicos tienen un gran peso a la hora de obtener un puesto laboral.
- El modelo creado en Python con ayuda de bibliotecas como PuLP, Scikit-learn, Pandas nos permitió encontrar una combinación con una efectividad de acuerdo a los datos de alrededor de 98.72% de éxito, lo que maximizo la obtención de un puesto laboral.
- Los resultados que se obtuvieron sugieren que, para maximizar la obtención de un puesto laboral, los estudiantes deben priorizar, tener un CGPA mayor o igual a 8, desarrollar habilidades comunicativas mayores o iguales a 7 y realizar 3 o más proyectos académicos para así maximizar su probabilidad de obtener un puesto laboral.
- Por último, pero no menos importante, gracias a la realización de este trabajo nos queda claro que la investigación de operaciones puede usarse no solo para maximizar unidades de una empresa o disminuir costos, sino que es aplicable a cualquier contexto con datos donde necesitemos maximizar algo, por medio de alguna combinación por lo tanto es aplicable a muchos problemas que van más allá de la producción.



## Referencias

Rodríguez, M., & Pérez, L. (2020). Factores determinantes en la empleabilidad de egresados universitarios. *Revista de Educación y Desarrollo*, 18(2), 45–58.

[https://www.scielo.org.mx/scielo.php?script=sci\\_arttext&pid=S2007-74672020000200127](https://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S2007-74672020000200127)

Universidad Abierta y a Distancia de México. (2020). Unidad 1: Programación lineal. Investigación de Operaciones.

[https://dmd.unadmexico.mx/contenidos/DCEIT/BLOQUE1/DS/05/DIOP/U1/descargables/DIOP\\_U1\\_Contenido.pdf](https://dmd.unadmexico.mx/contenidos/DCEIT/BLOQUE1/DS/05/DIOP/U1/descargables/DIOP_U1_Contenido.pdf)

Universidad San Ignacio de Loyola. (2019). Marco teórico: Programación lineal. Investigación de Operaciones.

<https://www.studocu.com/pe/document/universidad-san-ignacio-de-loyola/investigacion-de-operaciones/marco-teorico-programacion-lineal/38178693>

Quesada Ibargüen, V. M., Vergara Schmalbach, J. C., & Badrán Padauí, F. (Eds.). (2014). *Tópicos de investigación de operaciones*. Editorial Universitaria, Universidad de Cartagena.

<https://hdl.handle.net/11227/4836>

Supply Chain Data Analytics. (2023). *Modelado y resolución de problemas de optimización en Python*.

<https://www.supplychaindataanalytics.com/es/modelado-y-resolucion-de-problemas-de-optimizacion-en-python/>

Marionomics. (2023). *Tutorial para hacer programación lineal en Python con PuLP*.

<https://marionomics.com/tutorial-para-hacer-programacion-lineal-en-python-con-pulp/>

Islam, S. (2023). *College Student Placement Factors Dataset* [Dataset]. Kaggle.

<https://www.kaggle.com/datasets/sahilislam007/college-student-placement-factors-dataset/data>

Scikit-learn Developers. (s. f.). *scikit-learn: Machine learning in Python*. Scikit-learn.

<https://scikit-learn.org/stable/>

Pandas Development Team. (2024). *pandas: Python Data Analysis Library*.

<https://pandas.pydata.org/>

NumPy Developers. (2024). *NumPy*.

<https://numpy.org/>

## Anexos

### Anexo A. Código de Python para importar librerías, cargar datos, convertir columnas y configurar estilos para crear una grafica

```
# Importar librerías necesarias
import pandas as pd          # Para manipular y analizar los datos
import matplotlib.pyplot as plt # Para crear graficos basicos
import seaborn as sns        # Para una mejor visualizacion de los graficos

# Cargar y leer los datos del archivo CSV
df = pd.read_csv("college_student_placement_dataset_convert.csv")

# Convertir las columnas de 0/1 a 'Sí' y 'No'
# para que se entiendan mejor en las gráficas
df['Placement'] = df['Placement'].map({0: 'No', 1: 'Sí'})
df['Internship_Experience'] = df['Internship_Experience'].map({0: 'No', 1: 'Sí'})

# Configurar el estilos de los graficos
sns.set_theme(style="whitegrid")
```

### Anexo B. Código de Python para crear grafica de CGPA o promedio de calificaciones acumulativo

```
# Histograma de CGPA
# Le da un tamaño a la grafica
plt.figure(figsize=(6,4))
#La funcion histplot crea un histograma de los datos en CGPA
#El parametro bins=20 divide los resultados en 20 intervalos
# y KDE=True añade la curva de densidad
sns.histplot(df['CGPA'], bins=20, kde=True, color='skyblue')
#Se le da un titulo a la grafica y a los ejes
# y un tamaño de letra por medio de fontsize
plt.title('Distribución del CGPA', fontsize=14)
plt.xlabel('CGPA')
plt.ylabel('Cantidad de estudiantes')
# Muestra la grafica
plt.show()
```

### Anexo C. Código de Python para crear grafica de Grafico de barras de experiencia en practicas

```
# Gráfico de barras experiencia en prácticas
# Le da un tamaño a la grafica
plt.figure(figsize=(5,4))
# La funcion countplot crea un grafico de barras y
# cuenta cuántos registros hay por cada similitud en la columna Internship_Experience
# El parametro x cual es la columna que se quiere graficar
# El parametro data dice de donde se sacan los datos
# El parametro palette define los colores que se van a usar
sns.countplot(x='Internship_Experience', data=df, palette='pastel')
#Se le da un titulo a la grafica y a los ejes
# y un tamaño de letra por medio de fontsize
plt.title('Distribución de experiencia en prácticas', fontsize=14)
plt.xlabel('Experiencia en prácticas')
plt.ylabel('Cantidad de estudiantes')
# Muestra la grafica
plt.show()
```

#### Anexo D. Código de Python para crear grafica de Gráfico de barras de cantidad de estudiantes con un puesto laboral

```
# Cantidad de estudiantes con un puesto laboral
# Le da un tamaño a la grafica
plt.figure(figsize=(5,4))
# La funcion countplot crea un grafico de barras y
# cuenta cuántos registros hay por cada similitud en la columna Placement
# El parametro x cual es la columna que se quiere graficar
# El parametro data dice de donde se sacan los datos
# El parametro palette define los colores que se van a usar
sns.countplot(x='Placement', data=df, palette='pastel')
#Se le da un titulo a la grafica y a los ejes
# y un tamaño de letra por medio de fontsize
plt.title('Cantidad de estudiantes con un puesto laboral', fontsize=14)
plt.xlabel('Colocación laboral')
plt.ylabel('Cantidad de estudiantes')
# Muestra la grafica
plt.show()
```

#### Anexo E. Código de Python para crear grafica de dispersión de CGPA contra Habilidades comunicativas y coloreando si tienen o no un puesto laboral

```
# Dispersión CGPA contra Habilidades comunicativas
# y colores por colocación laboral
# Le da un tamaño a la grafica
plt.figure(figsize=(6,4))
# La funcion scatterplot crea un grafico de dispersion
# El parametro x cual es la columna que se quiere graficar en el eje x
# El parametro y cual es la columna que se quiere graficar en el eje y
# El parametro data dice de donde se sacan los datos
# El parametro hue dice de donde se sacan los datos para colorear en datos similares
# El parametro palette define los colores que se van a usar
sns.scatterplot(
    x='CGPA',
    y='Communication_Skills',
    data=df,
    hue='Placement',
    palette='coolwarm'
)
#Se le da un titulo a la grafica y a los ejes
# y un tamaño de letra por medio de fontsize
plt.title('Relación entre CGPA y habilidades comunicativas', fontsize=14)
plt.xlabel('CGPA')
plt.ylabel('Habilidades comunicativas')
plt.legend(title='Colocación')
# Muestra la grafica
plt.show()
```

## Anexo D. Código de Python utilizado para realizar el modelo de optimización para la obtención de un puesto laboral

```
# Importar librerías necesarias
import pandas as pd #Para manipular los datos
import numpy as np #Para operaciones matematicas
from sklearn.linear_model import LogisticRegression #Para encontrar los patrones en los datos
from sklearn.preprocessing import StandardScaler #Para estandarizar los datos y que estén en la misma escala
import pulp #Para poder encontrar la solución óptima

# Cargar los datos del archivo CSV
data = pd.read_csv('college_student_placement_dataset_convert.csv')

#Establecer limites para considerar un atributo como alto
cgpa_limite = 8.0 # CGPA minimo para considerarse dentro del rango
comms_limite = 7.0 # Puntuacion minima en habilidades comunicativas para considerarse dentro del rango
projects_limite = 3 # Numero minimo de proyectos que se puede considerar alto
extracurricular_limite = 7 # Puntuacion minima en actividades extracurriculares para considerarse dentro del rango

# Por medio de esto creamos nuevas columnas para cada uno de los atributos
# que nos dicen si el atributo es alto 1 o no 0
data['CGPA_Alto'] = (data['CGPA'] >= cgpa_limite).astype(int)
data['Habilidades_Comunicativas_Altas'] = (data['Communication_Skills'] >= comms_limite).astype(int)
data['Proyectos_Completados_Altos'] = (data['Projects_Completed'] >= projects_limite).astype(int)
data['Actividades_Extracurriculares_Altas'] = (data['Extra_Curricular_Score'] >= extracurricular_limite).astype(int)

# Definimos las variables sobre las cuales vamos a realizar la comparacion
#para encontrar la mejor combinacion de atributos
# X son las cualidades de los estudiantes 1 si la tienen y 0 si no
# Y es si consiguieron un puesto laboral 1 si y 0 no
X = data[['CGPA_Alto', 'Internship_Experience', 'Habilidades_Comunicativas_Altas', 'Proyectos_Completados_Altos', 'Actividades_Extracurriculares_Altas']]
y = data['Placement']

# Por medio de esto estandarizamos los datos
# para que esten en la misma sintonia y se puedan comparar
scaler = StandardScaler()
X_escalado = scaler.fit_transform(X)

# Por medio de esta funcion se realiza la regresion logistica o la importancia de cada factor
# y se obtienen los resultados
modelo = LogisticRegression()
modelo.fit(X_escalado, y)
coeficientes = modelo.coef_[0]

# Por medio de esto se convierten los resultados anteriores a porcentajes
# para saber que tanto importancia tienen para la obtencion de un puesto laboral
# y se asignan los resultados a cada variable
coeficientesNormalizados = np.abs(coeficientes) / np.sum(np.abs(coeficientes))
c1, c2, c3, c4, c5 = coeficientesNormalizados

# Por medio de la libreria pulp se crea el modelo de optimizacion
# se le da un nombre y se define que es de maximizar
prob = pulp.LpProblem("Maximize_Placement_Probability", pulp.LpMaximize)

# Definimos las variables de decision
# Como 0 y 1 para decir si el atributo es importante o no
x1 = pulp.LpVariable('CGPA_Alto', cat='Binary')
x2 = pulp.LpVariable('Experiencia_Practicas', cat='Binary')
x3 = pulp.LpVariable('Habilidades_Comunicativas_Altas', cat='Binary')
x4 = pulp.LpVariable('Proyectos_Completados_Altos', cat='Binary')
x5 = pulp.LpVariable('Actividades_Extracurriculares_Altas', cat='Binary')

# Se le indica al problema cual es la funcion que queremos maximizar
# que en este caso es la probabilidad de obtener un puesto laboral
prob += c1*x1 + c2*x2 + c3*x3 + c4*x4 + c5*x5, "Puntuacion_Empleabilidad"

# Se añaden todas las restricciones para que el modelo sea realista
# Carga académica menor o igual a 3: No más de 3 atributos altos
prob += x1 + x2 + x3 + x4 + x5 <= 3, "Restriccion_Recursos"
# Practicas respecto a CGPA: para realizar practicas requiere un CGPA alto
prob += x2 <= x1, "Experiencia_requiere_CGPA_Alto"
# Restricción mínima respecto a CGPA o Puntuación de Proyectos: Que el CGPA o proyectos sean altos
prob += x1 + x4 >= 1, "Minimo_Academico"

# Por medio de la funcion solve se resuelve el problema
prob.solve()

# Imprimimos el estado de la solución es decir si pudo encontrar una solución optima
print("Estado de la solución:", pulp.LpStatus[prob.status])
# Se imprimen los valores que el modelo encontro para cada variable 1 si es importante 0 si no
print("\nHabilidades para maximizar la obtencion de un puesto laboral:")
for variable in prob.variables():
    print(f"({variable.name}) = ({variable.varValue})")

# Se imprime el valor de la funcion objetivo
# es decir la probabilidad de obtener un puesto laboral en porcentaje
print(f"\nZ =: {pulp.value(prob.objective):.4f}")
print(f"Porcentaje de obtencion de un puesto laboral: {pulp.value(prob.objective) * 100:.2f}%")

# Interpretación de los resultados
print("\nRecomendacion para obtener un puesto laboral")
if x1.varValue == 1:
    print("Debe mantener un CGPA >= 8.0.")
if x2.varValue == 1:
    print("Debes buscar experiencia en practicas profesionales.")
if x3.varValue == 1:
    print("Debes desarrollar habilidades comunicativas >= 7.0.")
if x4.varValue == 1:
    print("Debes completar >= 3 proyectos academicos.")
if x5.varValue == 1:
    print("Debes participar en actividades extracurriculares y dar una puntuacion >= 7.")
```